APPENDIX A - Unitaxon Classification File Formats

This appendix documents the file format for a Unitaxon classification file and its addenda note file. This file format has a version number of 0.

Classification files should end with the suffix “.txn”. On Apple Macintosh systems, the file’s internal creator code should be ‘UTXN’ and its internal file type should be ‘TEXT’ if you wish to be able to double-click on the file and expect Unitaxon to be launched to read it.

User comment files for a given classification file should have the same name as the classification file, but instead of the “.txn” suffix, the comment file should have the suffix “.addenda”. On the Mac OS, the addenda file should also have the ‘UTXN’ and ‘TEXT’ creator/file type codes.

Both the main classification file and the added comments file are ASCII text files. All line feeds are ignored; carriage returns are treated as line ends.

Classification “.txn” file

Each “.txn” classification file is an ASCII text file, divided into sections. Sections within the text file are delimited by a single line beginning with the integer “-1”, or by the end of the file. Sections occur in a predefined order. This order and the format of subsequent sections is dependent on the version number specified in the first section.

Regardless of the type of section, the classification file consists of lines of text, where each line begins with an integer, each line represents a “record”, and various fields in each line’s record have tab-separated text values. There is a limit of approximately 4000 characters per line. The format of each logical line/record within a section is always the same, but different sections have different numbers of fields, with different meanings for the text in the fields. This format corresponds to what most generic database software uses to export database fields.

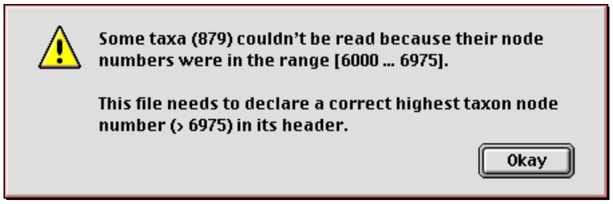
There are 12 text sections in the version 0 “.txn” file. In order, they are:

- Version
- Ranks
- Areas
- Times
- Parents
- Tree
- Seniors
- Commons
- Ranges
- Juniors
- Comments
- Notes

Version Section

The first line in the Version section contains two text fields, separated by a tab character. The first field is a non-negative version number (in this case version 0); the second field is a string of text representing the date associated with the creation/modification of this classification file. The second field doesn't have to be a date; it just needs to be a piece of text that identifies for the benefit of the end-user how to distinguish between this file and some later or earlier version with different data.

The second line similarly contains two text fields separated by a tab. The first field is a non-negative integer, and the second field is a short one-line title for the classification data. If the initial non-negative integer field is less than 100, it is ignored and an internal default is substituted. If it is 100 or more, then the Browser interprets it as a number that must be higher than the highest node number used to identify any senior synonym in the Tree Section data. Typically, taxon node numbers should be assigned consecutively, starting at 1, in which case this highest number is also the maximum number of taxa the browser can support while reading the file and building the taxonomic tree. If some taxon has an internal node number that is the same or higher than this number, it will be ignored and the Browser will report the problem when it has finished reading the file, in an alert such as this:



Some taxa (879) couldn't be read because their node numbers were in the range [6000 ... 6975].

This file needs to declare a correct highest taxon node number (> 6975) in its header.

This alert arose when the particular file being read declared 6000 taxa nodes, but the later Tree section had 879 node numbers that were higher than 6000. The fix would be to edit the ".txn" file and change the "6000" value on the second line of the Version section to something larger than 6975, such as "7000".

If the second line begins with an integer that is 100 or more, then the third line must be a sequence of eight tab-separated non-negative integer text fields. These fields provide a hint to the Browser of how many of various types of information to expect in the rest of the file. Field 1 declares how many junior synonyms to expect; field 2 declares how many common names to expect; field 3 declares how many range occurrences to expect; and field 4 declares how many notes and comments to expect; fields 5 through 8 are reserved and should be 0. Any field can be 0 which will cause the Browser to use internal default settings. These fields do not have to be completely accurate; the Browser can still read in the file, for instance, if the number of junior synonyms is more than the number declared here. It will just read the file in more slowly.

If the second line doesn't begin with an integer with value 100 or higher, then there is no third line declaring the eight non-negative integers (the Browser will use defaults).

The fourth "line" contains two text fields separated by a tab. The first field is a non-negative integer that is reserved and set to 0, and following the tab character is the start of a block of credits text which describes more fully what the classification data is about, who's responsible, how to get in touch, etc. All of this information can be continued on subsequent lines in the file, as long as no line begins with an integer (digit or minus sign). We recommend using no more than 10 lines of text.

The three pieces of text (data date, title, and credits) in the Version section are displayed verbatim in Unitaxon's **About this Data…** window.

The section is terminated by a line containing just the text "-1" at its start.

Ranks Section

The Ranks section defines the names of all taxonomic ranks used in this classification. Each rank name is assigned a rank code. The numerical rank codes, rather than the rank names, are recorded with each taxon in the Senior Tree Section later on.

Each line of text in the Ranks section has the same format. There are three tab-separated text fields per line/record.

The first field in a line is the positive integer index of the rank in the list of ranks. These index numbers should vary from 1 up to the number of rank names being defined. The index numbers define which ranks are subordinate to other ranks. Lower indices represent higher ranks in the classification.

The second field is the rank code, which should be a positive number, and unique among all defined ranks. Rank codes do not need to be consecutive or ordered. The rank codes (not the indices) are what are recorded with each individual senior synonym record in the Senior Synonym section later on.

The third field is the rank's name.

The section is terminated by a line containing just the text "-1" at its start.

Areas Section

The Areas section defines the names of areas used in the range data. Each area is assigned a code, and these codes are used later in the Range Section.

Each line of text in the Area section has the same format. There are two tab-separated text fields per line/record.

The first field is the area code, which should be a small positive integer, and unique among all defined areas in this section. The codes should begin with the number 1 and continue consecutively up to the number of areas.

The second field is the short abbreviated name for the area. These abbreviations are used when typesetting ranges as text, rather than using the interactive graphical display of an array of coordinates.

In Unitaxon Browser 2.0, the number of areas and the codes for each one are hard-wired, and the Area Section should contain 18 area definitions and look like this:

```
1      Af
2      Madagascar
3      Indian O
4      E. Indies
5      As
6      Mediterranean
7      Eu
8      Atlantic
9      Arctic O
10     N.A
11     Cent. A
12     W. Indies
13     S.A
14     Antarctica
15     New Zealand
16     Aus
17     New Guinea
18     Pacific
-1
```

The section is terminated by a line containing just the text “-1” at its start.

Times Section

The Times section defines the names of geologic epochs used in the range data. Each time period is assigned a code, and these codes are used later in the Range Section.

Each line of text in the Times section has the same format. There are two tab-separated text fields per line/record.

The first field is the period code, which should be a small positive integer, and unique among all defined time periods in this section. The codes should begin with the number 1 and continue consecutively up to the number of periods.

The second field is the short abbreviated name for the epoch. These abbreviations are used when typesetting ranges as text, rather than using the interactive graphical display of an array of coordinates.

In Unitaxon Browser 2.0, the number of epochs and the codes for each one are hard-wired, and the Times Section should contain 23 definitions and look like this:

```
1      L. Trias.
2      E. Juras.
3      M. Juras.
4      L. Juras.
5      E. Cret.
6      L. Cret.
7      E. Paleoc.
8      M. Paleoc.
9      L. Paleoc.
10     E. Eoc.
11     M. Eoc.
12     L. Eoc.
13     E. Olig.
14     L. Olig.
```

```
15     E. Mioc.
16     M. Mioc.
17     L. Mioc.
18     E. Plioc.
19     L. Plioc.
20     E. Pleist.
21     M. Pleist.
22     L. Pleist.
23     R.
-1
```

The section is terminated by a line containing just the text “-1” at its start.

Parents Section

The Parents section of the classification file declares the node numbers of all non-terminal taxa in the classification file, as well as the index of each taxon (terminal or non-terminal) in its parent’s list of subtaxa. This Section contains a subset of the same data in the Tree section. This subset needs to be read first in order to build the tree efficiently while allowing the taxon records in all the remaining sections to occur in any order.

Every taxon in the classification file has a unique, positive taxon node number. In addition, every taxon in the classification has a parent node number. The root of the classification (the taxon with the highest rank) logically has no parent in the file; for consistency in the data, however, this classification root will have a hidden stub-parent whose taxon node number is defined to be 0. All other parent node numbers refer to actual taxa in the data, and are positive. Ideally, they should begin with 1 and continue consecutively up to the number of taxa in the file. There can be gaps in the sequence, however.

Each line of text in the Parent section has the same format. There are two tab-separated text fields per line/record. Each line corresponds to one taxon in the tree.

The first field is a non-negative integer, which is the taxon node number for the parent of the taxon the line represents.

The second field is the index of this taxon in its parent’s list of subordinate (or children) taxa. The set of these indices for any given parent should be the integers 1 to N, where N is the number of children taxa the parent has.

The section is terminated by a line containing just the text “-1” at its start.

Tree Section

The Tree section of the classification file completes the declaration of the data’s tree structure, including senior synonym names and extinction data that are needed to display the tree. Some of the information is repeated from the previous Parents section.

Each line of text in the Tree section has the same format. There are five tab-separated text fields per line/record. Each line corresponds to a single taxon in the classification.

The first field is a positive integer representing the taxon's unique node number. These numbers typically start at 1 and are assigned consecutively as taxons are added to the classification. There can be gaps in the sequence, and they do not need to appear in order. As long as the node numbers are unique and no number is greater than or equal to the maximum node number declared in the Version section, they are legal.

There is no guarantee that any particular taxon will always be encoded with a given node number. It is entirely possible that two different updates of a given classification file will have different node numbers assigned for the same taxa. These numbers have meaning only within the confines of a single classification file. Unitaxon uses the node numbers as a shortcut when reading a classification file, but can then discard them or assign new node numbers as it sees fit.

The second field in the Tree section is a non-negative integer representing the node number of this taxon's parent taxon. The value 0 is reserved to be used as the pseudo-parent node number of the root taxon in the classification file. If any parent node number (other than 0) is not found as the value in the first field of some record in the Tree section, then something's internally inconsistent in the data. Typically, this second field is the same data as appears in the first field of the Parents section.

The third field is the positive index of this taxon in its parent's list of children taxa, where the first child has index 1. These indices are used to arrange the final order of every parent taxon's list of subordinate taxa in the tree.

The fourth field specifies whether the taxon is to be marked extinct or not. The field contains the text for a Boolean value. If the taxon is extinct, it contains either the four-character string "True" or the single character "T"; if not extinct, it contains either the five-character string "False" or the single character "F". These can also be specified in lower case characters as "t" or "f".

The final fifth field is a text field that contains this taxon's senior synonym name. If the taxon has a rank higher than Family, the name should be in all uppercase characters. Otherwise, only the first character in the name should be capitalized.

The section is terminated by a line containing just the text "-1" at its start.

Seniors Section

The Seniors section declares the remaining senior synonym data for each taxon in the tree. This includes its rank, citation, and several attributes.

Each line of text in the Seniors section has the same format. There are eight tab-separated fields per line/record. Each line corresponds to a single taxon in the classification.

The first field is the taxon's node number, a positive integer described in the Tree section above.

The second field is the taxon's rank code. These codes were declared earlier in the second field of the Ranks section. If the number found in this field was not declared in the Ranks section, then the data is inconsistent. This field is the rank code used to identify the rank, not the rank index used to order the ranks.

The third field is a text field that contains the author name for the senior synonym's bibliographic citation. The field can contain spaces, commas, quotes, or any other printable ASCII characters other than a tab character (which delimits the field).

The fourth field is a text field that contains text for the year of publication for the citation. Although the field is read as pure text, it should contain a usual four-digit year number.

The fifth field is a text field that specifies the page number or range of pages in the senior synonym bibliographic citation. Any printable ASCII characters other than a tab are legal.

The sixth field specifies whether the taxon is to be marked *incertae sedis* or not. The field contains the text for a Boolean value. If the taxon is *incertae sedis*, it contains either the four-character string "True" or the single character "T"; if not *incertae sedis*, it contains either the five-character string "False" or the single character "F". These can also be specified in lower case characters as "t" or "f".

The seventh field specifies whether the taxon is new. The field contains the text for a Boolean value. If the taxon is new, the field contains either the four-character string "True" or the single character "T"; if not new, the field contains either the five-character string "False" or the single character "F". These can also be specified in lower case characters as "t" or "f".

The eighth field specifies whether the taxon's rank is new. The field contains the text for a Boolean value. If the taxon is new, the field contains either the four-character string "True" or the single character "T"; if not new, the field contains either the five-character string "False" or the single character "F". These can also be specified in lower case characters as "t" or "f".

The section is terminated by a line containing just the text "-1" at its start.

Commons Section

The Commons section declares the common name or names for those taxa that have them.

Each line of text in the Commons section has the same format. There are four tab-separated fields per line/record. Each line corresponds to a single common name for some taxon. There can be more than one common name declared per taxon. Most extinct taxa do not have common names.

The first field is the node number of the taxon to which this common name belongs. The node number is a positive integer described earlier in the Tree section.

The second field is a positive integer that specifies the number by which this common name can be referenced elsewhere in the file if need be. This number should be unique among all common names in this section. Typically, these common name numbers are assigned consecutively starting from 1 to each common name as it was added to the master database. As with taxon node numbers, there is no guarantee that these numbers will remain assigned to the same common names among different updates of a given classification file.

The third field is the index of this common name in the taxon's list of common names. The first common name in the list has index 1, the second 2, etc.

The fourth field is a single common name for the taxon declared in the first field. Typically, it should not be capitalized, and can contain spaces, dashes, or other punctutation as necessary.

The section is terminated by a line containing just the text "-1" at its start.

Ranges Section

The Ranges section declares when and where and with what certainty a given taxon has occurred. This section is basically a list of coordinates in each taxon's range array. Every terminal taxon has to have a non-empty range array if it represents actual biological data. Non-terminal taxa generally do not have any range data assigned to them because Unitaxon builds synthetic ranges for them based on the ranges of their terminal taxa.

Each line of text in the Range section has the same format. There are five tab-separated fields per line/record. Each line corresponds to a single range occurrence in the range array for some taxon. There can be more than one occurrence declared per taxon.

The first field is the node number of the taxon to which this range occurrence refers. The node number is a positive integer described earlier in the Tree section.

The second field is the occurrence number. This is a unique positive number by which this range entry can be referenced elsewhere in this classification file. Typically, occurrence numbers are assigned consecutively starting at 1 to each new occurrence added to the master database; however, there can be gaps. As with taxon node numbers, there is no guarantee that these occurrence numbers will remain assigned to the same range occurrences among different updates of a given classification file.

The third field is a small positive integer that is the area number for the appropriate row in the range array. These numbers are declared in the Areas section earlier. The values of these numbers are hard-wired to be between 1 and 18 (inclusive), where 1 is the top row in the array, and 18 is the bottom row.

The fourth field is a small positive integer that is the time period number for the appropriate column in the range array. These numbers are declared in the Times section earlier. The values of these numbers are hard-wired to be betwen 1 and 23 (inclusive), where 1 is the left column in the array, and 23 is the right column (Recent).

The fifth field is a small positive integer having the value "1", "2", or "3". A value of 1 represents a "questionable" occurrence; a value of 2 represents a "possible" occurrence; and a value of 3 represents a "definite" occurrence for the given range's area/time coordinates.

The section is terminated by a line containing just the text "-1" at its start.

Juniors Section

The Juniors section declares the names, types, and citations for all junior synonyms in the classification. Every junior synonym entry in this section belongs to some taxon.

Each line of text in the Juniors section has the same format. There are nine tab-separated fields per line/record. Each line corresponds to a single junior synonym in the classification.

The first field is the node number of the taxon to which this junior synonym belongs. It is a positive integer described in the Tree section above.

The second field is the junior synonym number, a positive integer assigned to each junior synonym in this section and by which the junior synonym is referenced elsewhere in the file. The junior synonym number must be unique, and is typically assigned as junior synonyms are added to the master database, starting at 1. As with taxon node numbers, there is no guarantee that these junior synonym numbers will remain assigned to the same junior synonyms among different updates of a given classification file.

The third field is the index of this junior synonym its owning taxon's list of junior names. The first junior synonym has index 1, the second 2, etc.

The fourth field is a single character that declares the type of junior synonym. It can be one of three values: "O", "S", or "I", which respectively stand for Objective, Subjective, and Indeterminate.

The fifth field specifies whether the junior synonym is marked extinct or not. The field contains the text for a Boolean value. If the junior synonym is extinct, it contains either the four-character string "True" or the single character "T"; if not extinct, it contains either the five-character string "False" or the single character "F". These can also be specified in lower case characters as "t" or "f".

The sixth field is the junior synonym's name. If the taxon for which this is a junior synonym has a rank higher than the Family group, the name should be in all uppercase characters. Otherwise, only the first character in the name should be capitalized.

The seventh field is a text field that contains the name of the author of the junior synonym. The field can contain spaces, commas, quotes, or any other printable ASCII characters other than a tab character (which delimits the field).

The eighth field is a text field that contains text for the year of publication of the junior synonym. Although the field is read as pure text, it should contain a usual four-digit year number.

The ninth field is a text field that specifies the page number or range of pages on which the junior synonym was proposed. Any printable ASCII characters other than a tab are legal.

The section is terminated by a line containing just the text "-1" at its start.

Comments Section

The Comments section declares the comments for all senior synonyms that have them.

Each line of text in the Comments section corresponds to a single comment paragraph belonging to some taxon in the classification. Each line has the same format. There are two tab-separated fields per line.

The first field is the node number of the taxon to which this comment belongs. It is a positive integer described in the Tree section above.

The second field is the text of the comment paragraph, which can contain any printable ASCII characters. The rest of the line (paragraph text) can be up to approximately 4000 characters long; however, long paragraphs are discouraged in favor of several paragraphs.

For any given taxon node number, comment paragraphs are appended in the order they are encountered in the file.

If a line does not appear to begin with an integer, it is assumed to be a continuation of the comment text that ended the previous line, and it is assumed that the comment text had an embedded Return in it. That is, the comment text was more than one paragraph. Unitaxon will attempt to recover from this by creating and appending a new comment paragraph for the same taxon as the previous line.

The section is terminated by a line containing just the text “-1” at its start.

Notes Section

The Notes section declares all footnotes in the classification. Footnotes are attached to various pieces of a taxon’s data (not just senior synonyms) and are a more formal form of comment.

Each line of text in the Notes section corresponds to a single note paragraph belonging to some taxon, junior synonym, common name, or range occurrence in the classification. Each line has the same format. There are seven tab-separated fields per line. The first six fields are numbers, and the last field is the note paragraph text.

The first field is the node number of the taxon to which this note belongs. It is a positive integer described in the Tree section above.

The second field is the note number, a positive integer assigned to each note in this section and by which the note is referenced elsewhere in the file. The note number must be unique, and is typically assigned as note paragraphs are added to the master database, starting at 1. As with taxon node numbers, there is no guarantee that these note numbers will remain assigned to the same notes among different updates of a given classification file.

The third field is a non-negative integer that is either 0 or a junior synonym number declared in the Juniors section. If it is not 0, then the fourth and fifth fields of this record must be 0, and the note will be attached to the specified junior synonym, which should be a junior synonym of the taxon specified in the first field.

The fourth field is a non-negative integer that is either 0 or a range occurrence number declared in the Ranges section. If it is not 0, then the third and fifth fields of this record must be 0, and the note will be attached to the specified range occurrence, which should be a range occurrence of the taxon specified in the first field.

The fifth field is a non-negative integer that is either 0 or a common name number declared in the Commons section. If it is not 0, then the third and fourth fields of this record must be 0, and the note will be attached to the specified common name, which should be a common name of the taxon specified in the first field.

The sixth field is the note's superscript number. It should be a positive integer, unique among all notes for this taxon, and superscripts should start with 1.

The seventh field is the text of the note paragraph, which can contain any printable ASCII characters. The rest of the line (paragraph text) can be up to approximately 4000 characters long; however, long paragraphs are discouraged in favor of several paragraphs.

If a line does not appear to begin with an integer, it is assumed to be a continuation of the note text that ended the previous line, and it is assumed that the note text had an embedded Return in it. Unitaxon will attempt to recover from this by creating and appending a new note paragraph for the same object as was used to attach the note in the previous line.

The section is terminated by a line containing just the text "-1" at its start.

Classification ".addenda" files

All added comments to a classification are kept in its addenda file. This is a text file whose name has the same prefix as the classification file but which ends with ".addenda" as its suffix, rather than ".txn".

The addenda file contains ASCII text characters. Each line/record consists of tab-delimited fields. There is only one section, the Addenda section, delimited at its end by a line beginning with "-1".

Addenda Section

Each line of text in the Addenda section corresponds to a single added comment paragraph belonging to some taxon in the classification. The first line is a dummy, and serves only to declare the format of the rest of the lines. Each subsequent line has the same format.

There are seven tab-separated fields per line. The first field is a non-negative integer, and on the first line, this number declares the format version of the subsequent lines. Only format 0 is currently defined. The following describes the version 0 addenda file format.

On all subsequent lines, the first field is a non-negative integer. If the value of this number is 0, the rest of the fields will be parsed normally, but otherwise ignored. If the value is greater than 0, it is ignored and the rest of the fields are installed as an added comment. Typically, the value of this field is the record number, starting consecutively at 1.

The second field is the senior synonym name to which the comment is to be added. The addenda file does not use taxon node numbers, because these are transient and only defined within the main classification file. Since senior synonym names are unique, and because addenda files are much smaller than classificaiton files, it is more robust to attach the added comment by searching for the taxon name, rather than a possibly stale node number.

The third field is the added comment type. This is an integer, and should be set to “0”, the only currently defined type.

The fourth field is a reserved text field. It is ignored and should be set to empty. That is, there will be two tabs in a row delimiting the empty text field.

The fifth field is a text field that contains a short string specifying the author of the added comment. Typically, these will be the initials of whoever created the comment.

The sixth field is a text field that contains a short string specifying the date the added comment was created.

The seventh field is the text of the added comment, which can contain any printable ASCII characters. The rest of the line (paragraph text) can be up to approximately 4000 characters long; however, long paragraphs are discouraged in favor of several paragraphs.

If a line does not appear to begin with an integer, it is assumed to be a continuation of the added comment text that ended the previous line, and it is assumed that the comment text had an embedded Return in it. Unitaxon will attempt to recover from this by creating and appending a new comment paragraph after the previous comment paragraph.

The section is terminated by a line containing just the text “-1” at its start.