



# User's guide to the DELTA Editor

16 November 2016

**M.J. Dallwitz, T.A. Paine and E.J. Zurcher**

## Preface

If you are using DELTA for the first time, we strongly recommend that you *work through*, not just read, the examples in the sections 'Experimenting with the sample data' and 'Creating a dataset'. This will take 1–2 hours, and will save time in the long run.

## Abstract

The DELTA system is a flexible data-coding format for taxonomic descriptions, and an associated set of programs for producing and typesetting natural-language descriptions and keys, for interactive identification and information retrieval, and for conversion of the data to formats required for phylogenetic and phenetic analysis. This document is an introduction to the use of the DELTA Editor, a program for entering and maintaining DELTA data.



## Contents

<b>Introduction</b> .....	1
Notation .....	1
Font .....	1
<b>Experimenting with the sample data</b> .....	1
Opening the sample data .....	1
Making a copy of the data .....	2
Exporting to DELTA text files .....	2
Printing the character list .....	3
Producing natural-language descriptions .....	4
Generating a conventional key .....	4
Generating an interactive key .....	5
<b>Creating a dataset</b> .....	6
Developing a character list: reproducibility .....	6
Starting a new dataset .....	6
Entering characters .....	7
Saving the dataset .....	8
Entering taxon names .....	9
Recording simple attributes of taxa .....	10
Exporting to DELTA text files .....	10
Producing output via action sets .....	11
<b>Importing DELTA text files</b> .....	15
<b>Saving or exporting a dataset</b> .....	16
<b>Views of the data</b> .....	16
<b>The attribute editors</b> .....	17
The tree-view attribute editor .....	17
The grid-view attribute editor .....	18
Using the 'Enter' key in attribute editing .....	19
<b>The item editor</b> .....	20
<b>The character editor</b> .....	21
The state editor .....	22
The notes editor .....	22
The image-association editor .....	23
The controlling-attribute editor .....	23
The 'controlled-by' chooser .....	24
<b>The image settings dialog</b> .....	25
<b>Image windows and overlays</b> .....	26
<b>Entering special symbols</b> .....	29
<b>Planned enhancements and known bugs</b> .....	31
<b>Appendix 1. Hints on organization and maintenance of data</b> .....	32
<b>Appendix 2. Why must character states be mutually exclusive?</b> .....	35
<b>Acknowledgements</b> .....	37
<b>References</b> .....	37
<b>Citation</b> .....	38



## Introduction

This document is an introduction to the use of the DELTA Editor 'Delta', a program for entering and maintaining DELTA (Description Language for Taxonomy) data. The DELTA Editor will eventually replace and extend the functionality of the programs Confor (data translation) and Delfor (data maintenance). The program is still being developed, and many planned features have not yet been implemented.

This version provides for the editing, integrity checking and maintenance of DELTA datasets (including directives files), but will not perform any translations of the data. Hence it will have to be used in conjunction with Confor. However, runs of Confor and other DELTA programs can be initiated from within the Editor.

This document should be used in conjunction with the [User's guide to the DELTA System](#), which contains details of the purpose and syntax of directives used by Confor and other DELTA programs.

You should back up your data regularly. For additional security, export the data to DELTA text files (see [Saving or Exporting a Dataset](#)), and back up these as well as the binary database file (.dlt).

### Notation

It is often necessary to describe a sequence of selections in menus and dialogs. We will represent such sequences by a list of the selections, separated by the symbol '>'. Unless otherwise indicated, the first selection is in the main menu bar of the program. For example, 'File > New Dataset' means 'in the main menu bar, click on 'File'; then, in the menu that is displayed, click on 'New Dataset'.

### Font

The font used in the display can be set via 'Window > Set Font'.

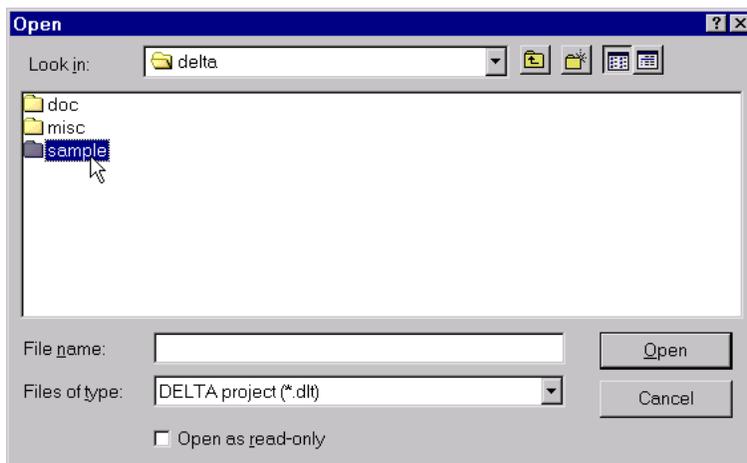
## Experimenting with the sample data

Before starting to prepare your own data, we strongly recommend *working through* (not just reading) the following examples, which use the sample data that are supplied with the DELTA programs. This should take less than an hour, but will save time in the long run.

### Opening the sample data

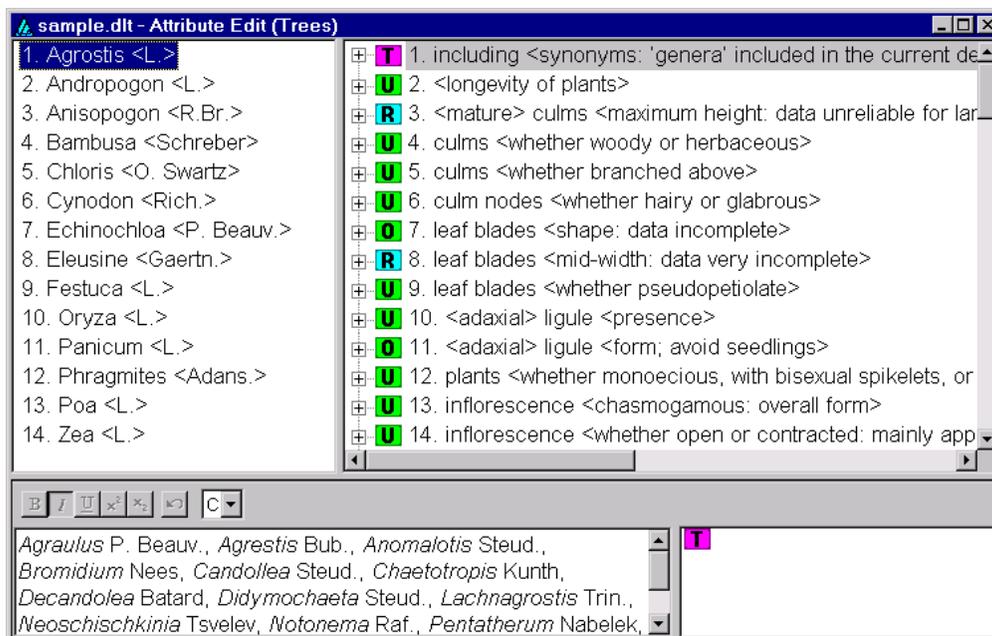
To start the DELTA Editor, click the Windows 'Start' button, and in the menus that are displayed, select 'Programs > Delta > Delta Editor'.

The program window is initially empty, except for a menu bar at the top. To open the sample data, select 'File > Open Dataset'. The following dialog is displayed. (If a different folder is shown, browse to the folder in which the program is installed.)



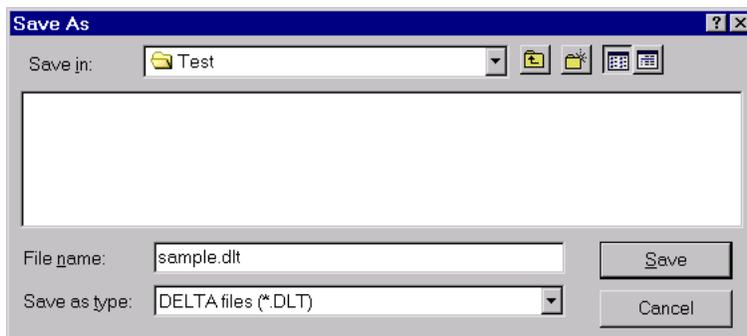
Click on 'sample', and click 'Open'. In the new dialog that is displayed, click on 'sample.dlt', and click 'Open'.

The following screen is displayed. It is described in detail later (see [The tree-view attribute editor](#)).



## Making a copy of the data

So that you can experiment freely with the sample data without the risk of corrupting the original files, make a copy as follows. Select 'File > Save Dataset As'. The 'Save As' dialog is displayed. You are in the 'sample' folder. Click ('Up One Level') to take you to the Delta folder, then click ('Create New Folder'). Type 'Test', click on the adjacent 'folder' icon to select the newly created folder, and click 'Open'. The following dialog is displayed.

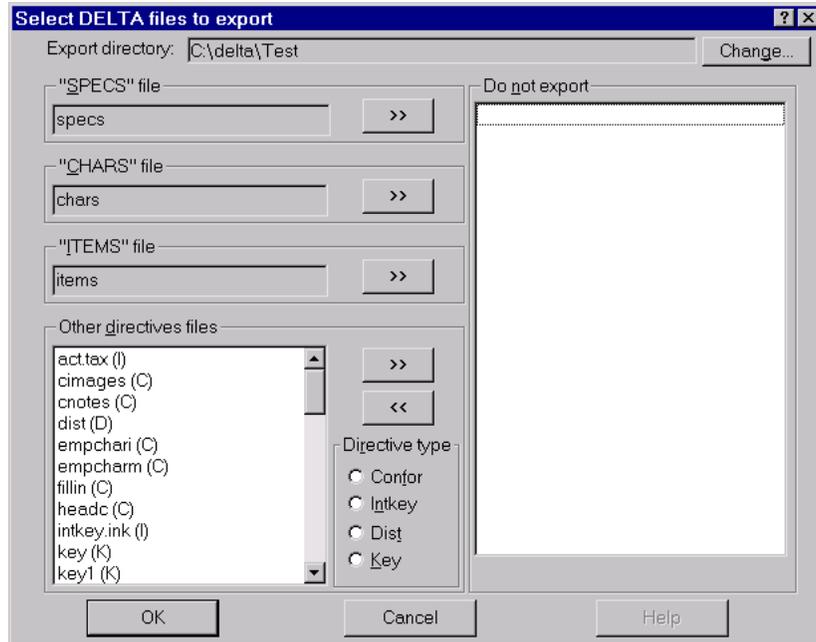


Check that the 'Save in' folder is 'Test' (as shown). Then type 'Testdata' in the 'File name' box (overwriting 'sample.dlt'), and click 'Save'.

## Exporting to DELTA text files

Before the data can be used by other programs such as Confor, Key, etc., they must be 'exported' to text files. These comprise data files in the DELTA format (e.g. 'specs' — specifications; 'chars' — character list), and 'directives' or 'action' files (e.g. 'tonatr' — translate into natural language), which specify how the data are to be processed. These files are described in detail in the [User's Guide to the DELTA System](#).

To export the data to DELTA text files, select 'File > Export Directives'. The following dialog is displayed.

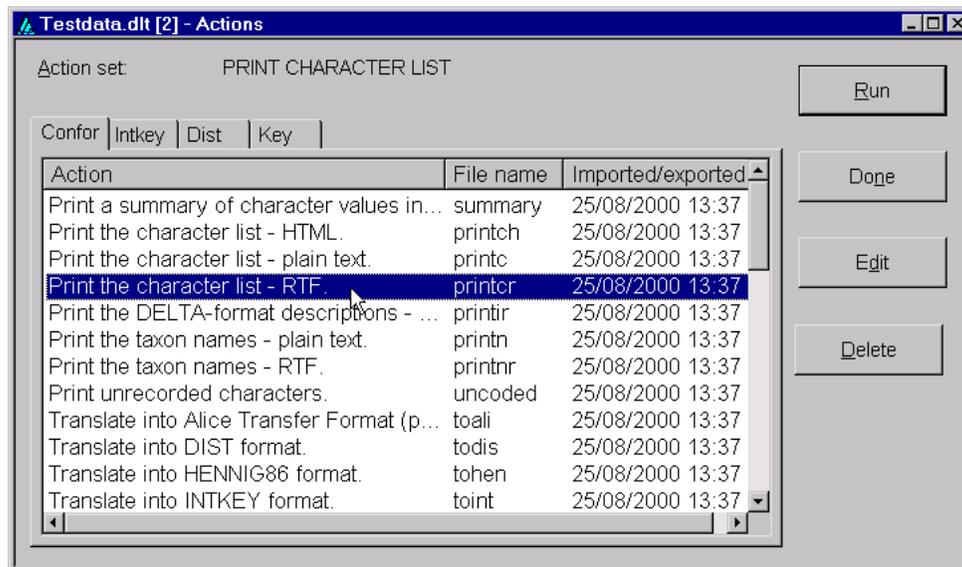


Click 'OK'. An 'Export Status' window shows progress. When the process is finished, click 'Done'. This reveals the 'Export Log' window; close it by clicking **X**.

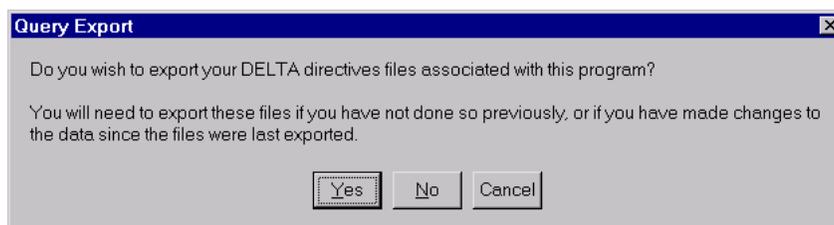
Note that the database stores information about the exporting. Therefore, when you close the Editor later, you will be asked whether you want to save the changes, even if you have not explicitly altered the data. You should normally answer 'Yes'.

## Printing the character list

To format and print the character list, proceed as follows. Select 'View > Action Sets'. The following dialog is displayed (you will see different dates and times in the 'Imported/exported' column).



Select 'Print the character list - RTF.', as shown ('RTF' is Microsoft's 'Rich Text Format'). Click 'Run'. The following message is displayed.



As you have already exported all the files, click 'No'. Confor is then run, and a log of the run is displayed. This displays the names of the output files, 'printer.lst' and 'rtfchars.rtf', and would also show any error messages. Also displayed is a message asking whether you want to close the window (i.e. the log); click 'Yes'.

'printer.lst' is a text file containing a more detailed log of the process. There is usually no need to look at it, but it may occasionally be helpful to do so if there are errors, or if the output is not what you expected.

'rtfchars.rtf' is an RTF file containing the main output. The name indicates that the file 'chars.rtf' is in the sub-folder 'rtf' of the 'Test' folder. You can open and print 'chars.rtf' in MS Word, MS Word Viewer, or WordPad. **MS Word Viewer** can be obtained free of charge from Microsoft. WordPad comes with Windows, but its formatting capabilities are limited.

N.B. If a file is open in Word, no other program can use it. Therefore, if you are using Word to view chars.rtf, you must close the file before running 'printer' again to regenerate it.

## Producing natural-language descriptions

If you have closed the 'Actions' dialog, open it again by selecting 'View > Action Sets'. There are four options for translating into natural-language descriptions.

1. HTML — Hypertext Markup Language. This is a format used for Web pages. Each taxon description is in a separate file, and an index to these files is produced.
2. Plain text.
3. RTF, separate file for each taxon. These files are usually used for displaying the descriptions within the interactive-key program, Intkey.
4. RTF, single file for all taxa. This is the best form for printing the entire set of descriptions.

Click on the first option, and click 'Run'. This creates a file 'index.htm' in the 'Test' folder, and files 'agrostis.htm', etc., in the subfolder 'www'. Open 'index.htm' in a Web browser such as Firefox or Internet Explorer. The links to the descriptions should work, but other links, such as those to the Introduction ('intro.htm') and the images, will not work, because those files are not an integral part of the DELTA database, and so were not copied to the 'Test' folder by the process that copied the DELTA files.

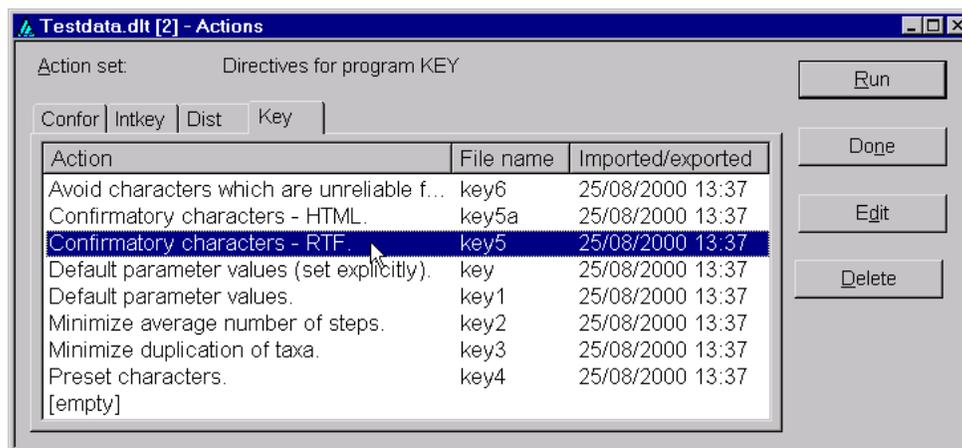
(If you wish, you can use Windows Explorer to copy the files necessary to make the links work. Create a folder 'images' in the 'Test' folder, and copy to it all the files in the 'sample\images' folder. Copy all the '.htm' files from the 'Sample' folder to 'Test\Www'. Copy 'deltlogo.gif' and 'style.css' (if present) from the 'Delta' folder to 'Test' and 'Test\www'. To make the link to the character list work, run the Action Set 'Print the character list – HTML'.)

## Generating a conventional key

Confor is the only one of the programs that works directly from the DELTA-format data files. The other programs, Intkey, Dist, and Key work from special data files, which are created by Confor from the DELTA data files.

If you have closed the 'Actions' dialog, open it again by selecting 'View > Action Sets'. Select 'Translate into KEY format', and click 'Run'. This creates data files 'kchars' and 'kitems', which are used by the key-generation program, Key.

In the 'Actions' dialog, click on the 'Key' tab. The following dialog is displayed.



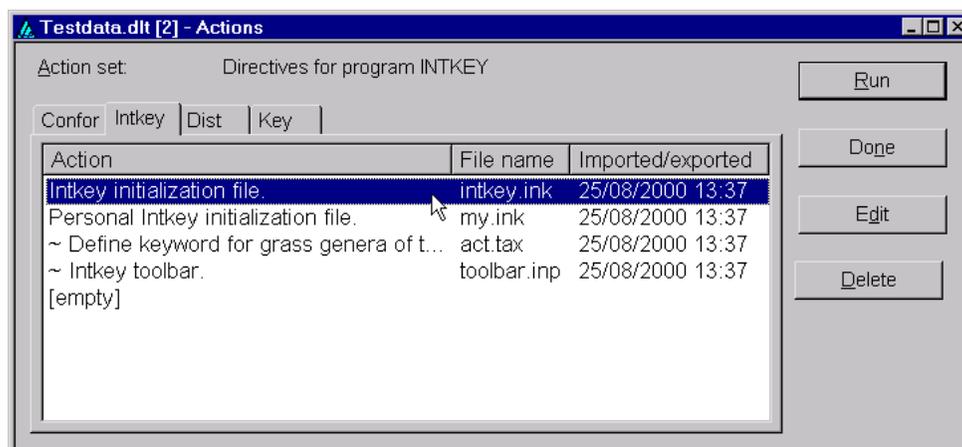
Select 'Confirmatory characters - RTF' (as shown), and click 'Run'. The key is created in the file 'key5.rtf' in the 'Test\rtf' folder. You can view and print it using MS Word or an equivalent program, as described previously for the character list.

If you want to create another key, using one of the other 'Key' action sets, there is no need to run 'Translate into KEY format' again. (It would be necessary, however, if you changed the data, or the parameters that are defined in the 'Translate into KEY format' action set. More details about this are given later.)

## Generating an interactive key

If you have closed the 'Actions' dialog, open it again by selecting 'View > Action Sets'. If necessary, select the 'Confor' tab. Select 'Translate into INTKEY format', and click 'Run'. This creates data files 'ichars' and 'iitems', which are used by the interactive-key program, Intkey.

In the 'Actions' dialog, click on the 'Intkey' tab. The following dialog is displayed.



Select 'Intkey initialization file' (as shown), and click 'Run'.

Intkey has built-in help. To get started, run Intkey and select 'Help > Introduction'. There is an extensive example in the [Overview of the DELTA System](#).

When you close Intkey, you will be given the opportunity to add the dataset to the Intkey index.

You can also run Intkey directly, without using the DELTA Editor. Click the Windows 'Start' button, and in the menus that are displayed, select 'Programs > Delta > Intkey'. A dialog for selecting a dataset is displayed. If you previously added the dataset to the Intkey index, you can now select it in the 'Select by title' box. Otherwise, use the 'Select by name of initialization file' box; click the 'Browse' button and find the file 'intkey.ink' in the 'Test' folder.

## Creating a dataset

### Developing a character list: reproducibility

If you are starting a dataset from scratch, begin by entering only a few characters and a few taxa (say about 5 of each), and recording the data for these taxa. Then test *all* the applications you intend to use — for example, produce natural-language descriptions, a conventional key, an interactive key, and a cladistic tree. Then add a few more characters and taxa, and repeat the testing. This iterative procedure helps you develop strategies for character definition (e.g. wording, number of states, multistate versus numeric, arrangement in sentences), which you can apply and refine as development of the character list proceeds.

Ideally, after the above process, the character list should be tested by having several people independently record descriptions of about 10 disparate taxa. The different versions of the descriptions will probably be different, i.e. the results will not be reproducible. You need to detect such problems before you have recorded much data.

The Differences option in Intkey can easily pinpoint the discrepancies. The reasons for them can then be discussed, and the character list refined on this basis. This process should then be iterated, adding a few more taxa each time, until the character list seems satisfactory.

Character notes, and preferably character illustrations, should be added during the development of the character list. There is a tendency to think that these are only for the benefit of the end user, and to postpone adding them. However, they are an essential part of obtaining reproducible descriptions. Without precise character notes, the character concepts of even the principal author may drift during the recording of the data, as more specimens are seen. Then, data recorded near the end of the project may not be comparable to data recorded near the beginning. Pay particular attention to the boundaries between character states, i.e. describe and illustrate intermediate conditions, and say to which state they are deemed to belong (unless you intend recording such intermediates as belonging to both states).

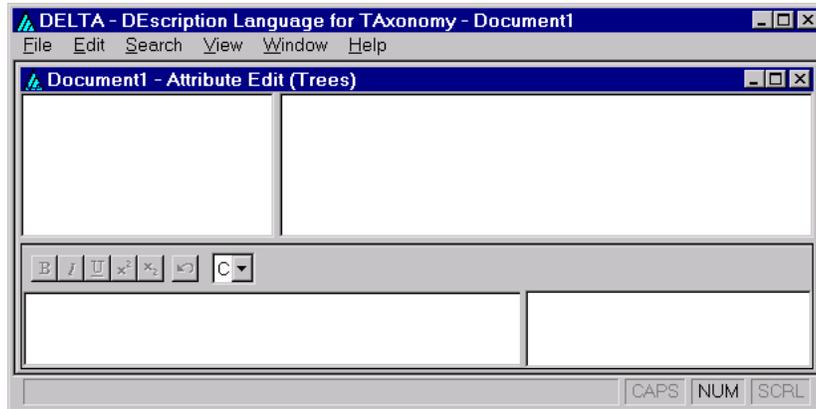
You should also record your *reasons* for defining the characters as you have, while these reasons are still fresh in your mind. Otherwise, other taxonomists (or even you yourself, later) may fail to fully appreciate

the definitions, and change them, to the detriment of your data and possibly to the detriment of future data recording.

## Starting a new dataset

To start the DELTA Editor, click the Windows 'Start' button, and in the menus that are displayed, select 'Programs > Delta > Delta Editor'.

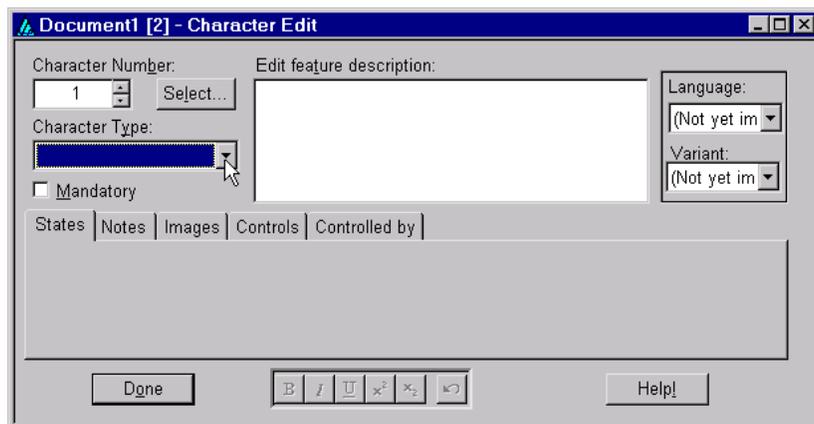
The program window is initially empty, except for a menu bar at the top. To start a new dataset, select 'File > New Dataset'. The 'Attribute Edit (Trees)' window is displayed.



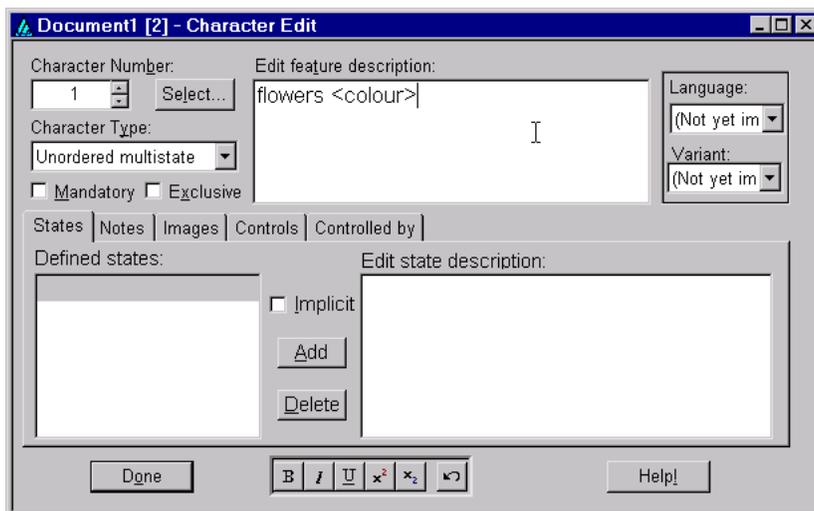
You can enter data either by typing it into the Editor or by importing existing DELTA text files (see [Importing DELTA Text Files](#)).

## Entering characters

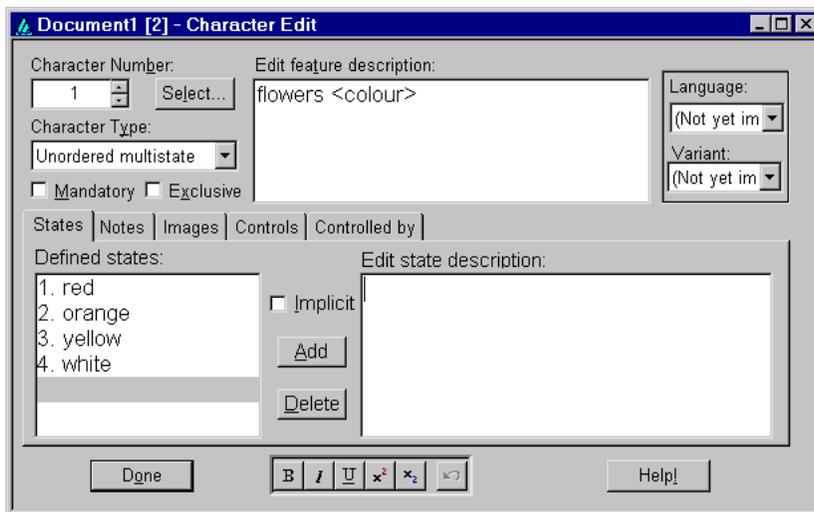
You can enter character descriptions either by choosing 'View > Character editor' from the menu, or by right-clicking the mouse in the upper right pane of the data window and selecting 'Append new character to end of list' from the menu that will appear. The 'Character Edit' dialog is displayed.



As an example, we will enter 2 characters — for flower colour and leaf length. In the 'Character Type' box, click on ▾ and select 'Unordered multistate'. In the 'Edit feature description' box, enter 'flowers <colour>' (N.B. no capital letter). The window now looks like this.

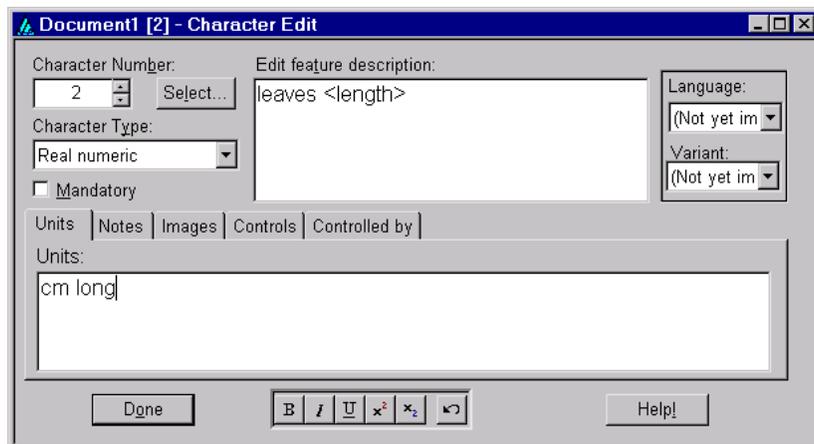


Click in the 'Edit state description' box, type 'red', and press 'Enter'. Type 'orange', and press 'Enter'. Similarly, enter states 'yellow' and 'white'. The window now looks like this.

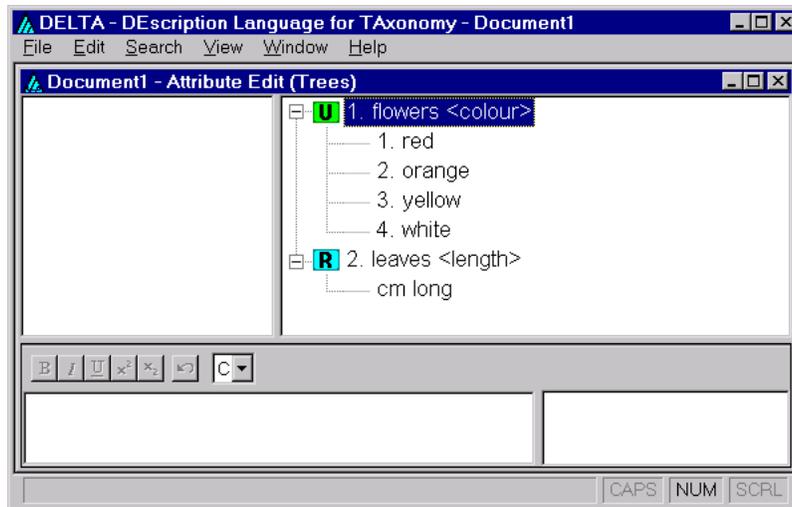


In some contexts, characters are selected from lists of their 'feature descriptions' alone (i.e. without their states). The word 'colour' is necessary to allow the character to be distinguished in these contexts from other flower characters, such as 'size' and 'number of petals'. Placing the word in angle brackets causes it to be omitted from descriptions, e.g. 'flowers orange' instead of 'flowers colour orange'.

In the 'Character number' box, click . This changes the character number to 2. In the 'Character Type' box, select 'Real numeric'. In the 'Edit feature description' box, enter 'leaves <length>'. In the 'Units' box, enter 'cm long'. The window now looks like this.

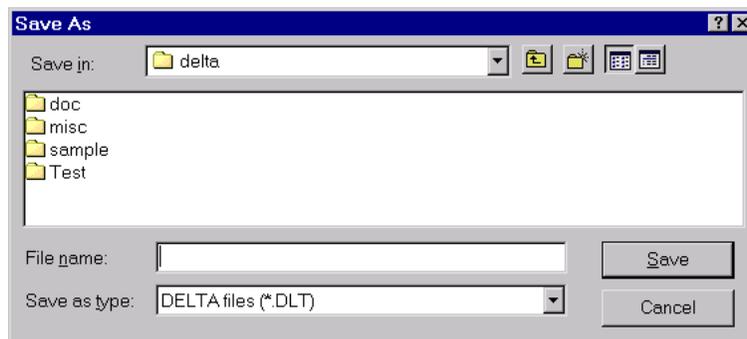


Close the 'Character Edit' window by clicking 'Done'. Right click in the top-right pane of the 'Attribute Edit' window, and, in the menu that appears, select 'Expand all'. The window now looks like this.

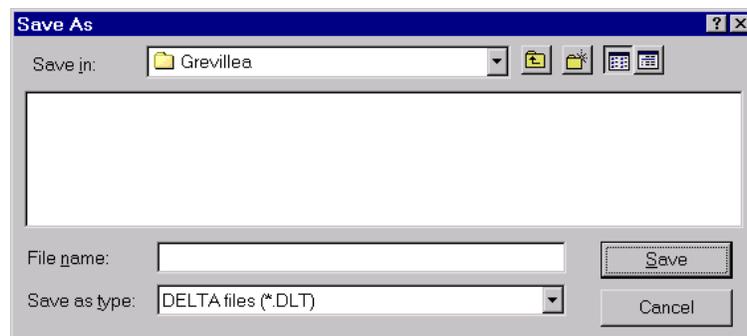


## Saving the dataset

Select 'File > Save Dataset'. The following window is displayed.



Create a new folder and move to it as follows. Click . Type a suitable name, e.g. 'Grevillea'. Click on the  icon adjacent to the newly created folder, causing it to be selected. Click 'Open'. The dialog should now look like this.

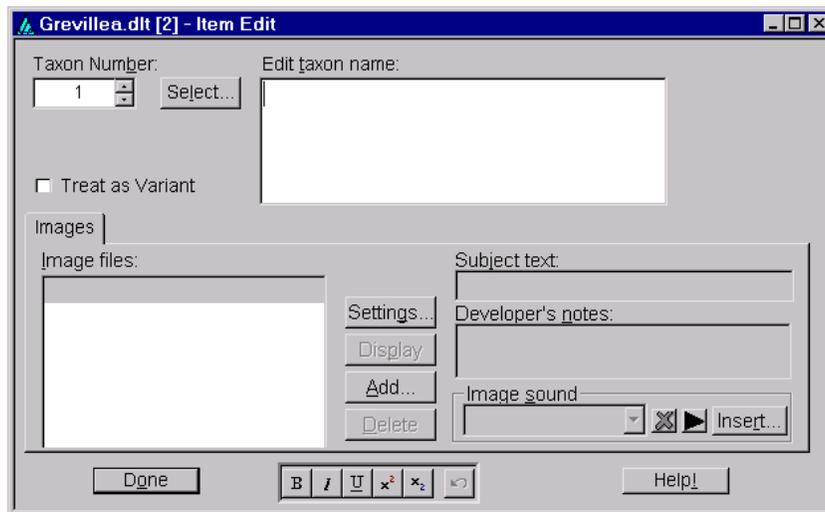


Make sure that the 'Save in' folder is the newly created one ('Grevillea'). Type a suitable name, e.g. 'Grevillea', in the 'File name' box, and click 'Save'. This saves the data in the file 'Grevillea.dlt'. This is a binary file — it will be intelligible only when viewed in the DELTA Editor.

*It is important to save each dataset in a separate folder, as described above.* Later on, you will need to produce many other files associated with the dataset, and it is convenient to give these the same names in every dataset. Thus, to keep the files distinct, they must be in different folders.

## Entering taxon names

You can enter new taxon names either by choosing ‘View > Taxon editor’ from the menu, or by right-clicking the mouse in the upper left pane of the data window, and selecting ‘Append new taxon to end of list’ from the menu that will appear. The ‘Taxon Edit’ dialog is displayed.



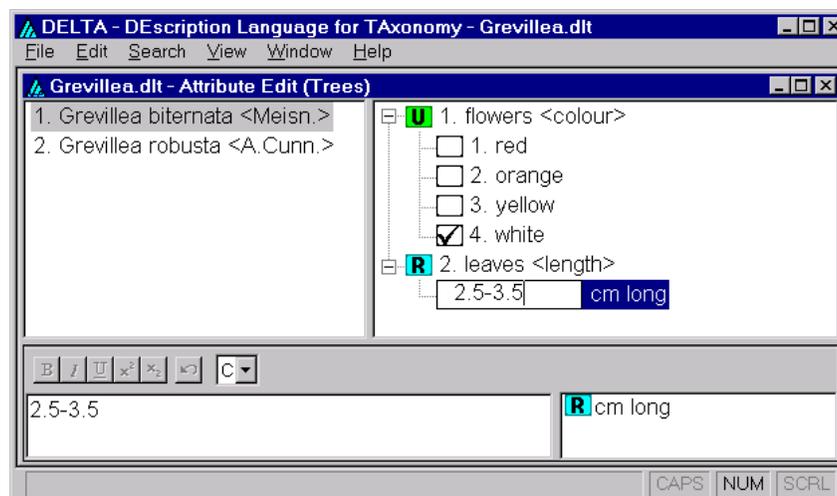
As an example, we will enter 2 taxon names. In the ‘Edit taxon name’ box, type ‘*Grevillea biternata* <Meisn.>’. Use the **I** button at the bottom of the window to produce italics. Putting the authority in angle brackets as shown causes it to be omitted in certain contexts, for example, in keys.

In the ‘Taxon number’ box, click **▲**. This changes the taxon number to 2. Enter the second taxon name, ‘*Grevillea robusta* <A.Cunn.>’. Then click ‘Done’.

The ‘Attribute Edit (Trees) window is again displayed. The first taxon, *Grevillea biternata*, is selected, and the characters in the top-right pane now have boxes for recording the character values for this taxon.

## Recording simple attributes of taxa

To record the attribute ‘flowers white’ for the selected taxon, *Grevillea biternata*, click in the box adjacent to the character state ‘white’ in the top-right pane. A tick appears in the box. To record the attribute ‘leaves 2.5–3.5 cm long’, type ‘2.5–3.5’ in the box under ‘leaves <length>’. The window now looks like this.



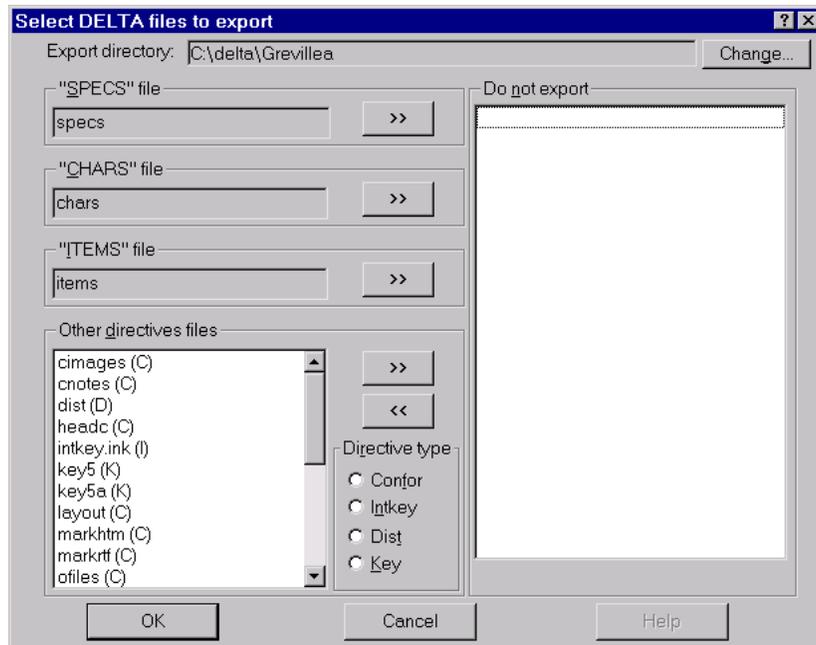
Now select the second taxon, *Grevillea robusta*, and record ‘flowers orange’, and ‘leaves 20–30 cm long’. Save the file (‘File > Save Dataset’).

For instructions on entering more complex attributes, see [The tree-view attribute editor](#).

## Exporting to DELTA text files

Before the data can be used by other programs such as Confor, Key, etc., they must be 'exported' to text files. These comprise data files in the DELTA format (e.g. 'specs' – specifications; 'chars' – character list, 'items' – item (taxon) descriptions), and 'directives files' or 'action sets' (e.g. 'tonatr' – translate into natural language), which specify how the data are to be processed. These files are described in detail in the [User's Guide to the DELTA System](#).

To export the data to DELTA text files, select 'File > Export Directives'. The following dialog is displayed.



Click 'OK'. An 'Export Status' window shows progress. When the process is finished, click 'Done'. This reveals the 'Export Log' window; close it by clicking .

Note that the database stores information about the exporting. Therefore, when you close the Editor later, you will be asked whether you want to save the changes, even if you have not explicitly altered the data. You should normally answer 'Yes'.

## Producing output via action sets

Action sets are sets of instructions that will be read and processed by various programs, in order to produce output such as natural-language descriptions, or to carry out pre-defined operations in the interactive identification program, Intkey. Templates for various commonly used action sets are automatically included when a new dataset is created as described above. These templates are based on a subset of the action sets included in the sample data supplied with the programs (see [Experimenting with the sample data](#)). The sample data contain references to taxon names and to character and state numbers that will usually be inconsistent with or inappropriate for another dataset. Rather than deleting directives containing such information, they have been turned into comments, in order to serve as examples.

The program Confor converts DELTA text files (see [previous section](#)) into natural-language descriptions, or into the formats required by other programs such as Intkey (interactive key), Dist (generating distance matrices), and Key (generating conventional keys) (see [Experimenting with the sample data](#)).

Confor action sets are based on the directives

- \*INPUT FILE specs
- \*INPUT FILE chars
- \*INPUT FILE items

Although this sequence of directives constitutes a valid action set, it only reads and checks the basic DELTA text files: the specifications, characters, and item (taxon) descriptions. To produce output, we

need to add directives specifying the type of output required, and the names of the output files. For example, the action set

```
*INPUT FILE specs
*TRANSLATE INTO NATURAL LANGUAGE
*PRINT FILE descrip.txt
*INPUT FILE chars
*INPUT FILE items
```

produces natural-language descriptions in a file descrip.txt.

Further directives can be added to define the required output in more detail, and to document the purpose of the action set and individual directives. For example, the following action set causes new paragraphs to be started at designated points in the descriptions.

```
*COMMENT Translate into natural language - plain text
*INPUT FILE specs
*TRANSLATE INTO NATURAL LANGUAGE
*PRINT FILE descrip.txt
*COMMENT Start new paragraphs at the designated points.
*NEW PARAGRAPHS AT CHARACTERS 1-2 12 25-26 68 77-78 87-89
*INPUT FILE chars
*INPUT FILE items
```

The COMMENT directives are skipped by the program – they are for the benefit of human readers.

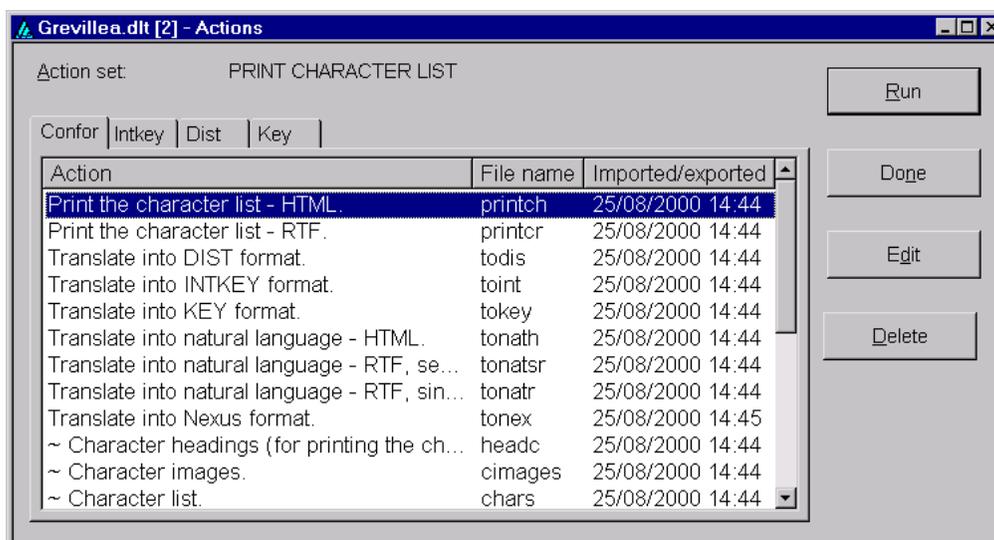
The first two of the above action sets will work with any datasets. The third will fail if the dataset does not have at least 89 characters (and, of course, the positions of the new paragraphs are unlikely to be suitable for other datasets). The action set will be suitable for general use again if we deactivate the NEW PARAGRAPHS AT CHARACTERS directive by turning it into a comment, thus:

```
*COMMENT : NEW PARAGRAPHS AT CHARACTERS 1-2 12 25-26 68 77-78 87-89
```

This allows the contents of the original directive to remain as an example. In the templates, the directives that are specific to the sample data, and hence unsuitable for general use, are deactivated in this way. The colon after COMMENT has no significance for the program; we use it in the templates so that deactivated directives can be more easily distinguished from ordinary comments. The user can edit the list of characters to suit the requirements of another dataset, then reactivate the directive by removing ‘COMMENT :’, for example:

```
*NEW PARAGRAPHS AT CHARACTERS 1 20 34
```

To view a list of action sets, select ‘View > Action Sets’. The following ‘Actions’ dialog is displayed.

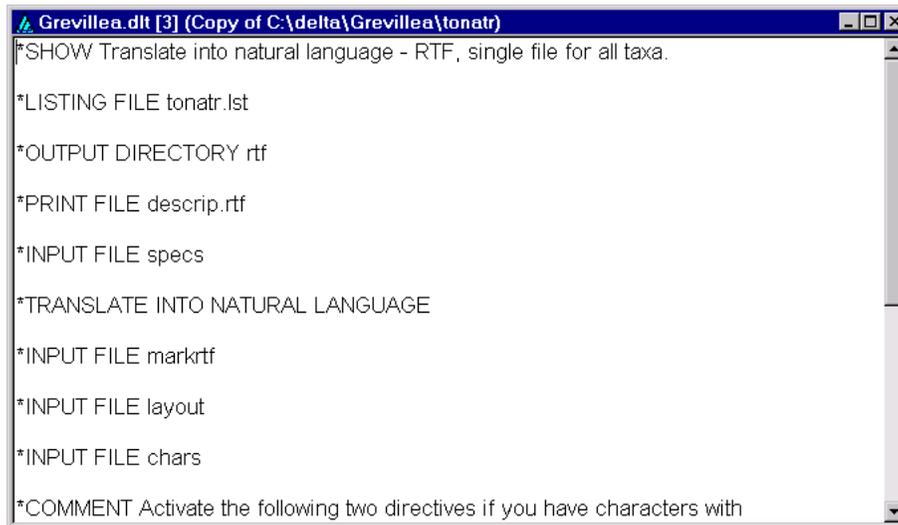


The action-set descriptions shown in the column labelled ‘Action’ are taken from a ‘COMMENT’ or ‘SHOW’ directive at the start of each action set. The text in the ‘File name’ column is the internal file name of the action set, and also the name of the corresponding external file when the data are exported to

DELTA text files. The 'Imported/exported' column show the date and time when the action set was most recently imported from or exported to its external counterpart.

The sets whose descriptions start with '~' are partial sets, which are incapable of producing output by themselves, but are invoked by other action sets.

Click on 'Translate into natural language – RTF, single file for all taxa.', and then click 'Edit'. The following window is displayed.



```

*SHOW Translate into natural language - RTF, single file for all taxa.

*LISTING FILE tonatr.lst

*OUTPUT DIRECTORY rtf

*PRINT FILE descrip.rtf

*INPUT FILE specs

*TRANSLATE INTO NATURAL LANGUAGE

*INPUT FILE markrtf

*INPUT FILE layout

*INPUT FILE chars

*COMMENT Activate the following two directives if you have characters with

```

Note the 'INPUT FILE' directives, which invoke other files of directives, such as 'layout'. Close this window by clicking . The 'Actions' dialog is revealed again. Scroll down and click on '~ Layout for natural-language descriptions.', and click 'Edit'. The following window is displayed (you will see some additional blank lines, which have been removed here to reduce the size of the image).



```

*COMMENT ~ Layout for natural-language descriptions.

*REPLACE ANGLE BRACKETS
*OMIT CHARACTER NUMBERS
*OMIT INNER COMMENTS
*OMIT INAPPLICABLES

*COMMENT Specify slot in which image links will appear in HTML descriptions.
There must be a corresponding character in the character list, and no data
should be recorded against this character in the item descriptions (the data
come from the TAXON IMAGES directive in the file 'images').
*COMMENT : CHARACTER FOR TAXON IMAGES 88

*COMMENT : EXCLUDE CHARACTER 89

*COMMENT : NEW PARAGRAPHS AT CHARACTERS 1-2 12 25-26 68 77-78 87-89

*COMMENT Group characters into sentences.
*COMMENT : LINK CHARACTERS 3-5 7-9 10-11 13-15 16-17 18-21 26-29 32-37 41-42
45-47:52-54 48-51 55-56 57-59 62-63 64-65 68-70 71-72 73-75 78-84

*COMMENT Headings within in taxon descriptions. The numbers specify the
character before which the heading is placed.
*COMMENT : ITEM SUBHEADINGS
#2. \b{}Habit, vegetative morphology\b0{}.
#12. \b{}Inflorescence\b0{}.
#25. \b{}Female-sterile spikelets\b0{}.
#26. \b{}Female-fertile spikelets, florets, fruit\b0{}.
#68. \b{}Photosynthetic pathway, leaf blade anatomy\b0{}.
#77. \b{}Special diagnostic feature.\b0{}.
#78. \b{}Taxonomy, distribution\b0{}.
#87. \b{}Anatomical references\b0{}.
#88. \b{}Illustrations\b0{}.

```

The directives that start ‘\*COMMENT : ’ have been deactivated for the reasons given above. They can be activated by removing the text ‘COMMENT : ’, but the Editor will reject the changed action set when you attempt to save it, unless appropriate changes have also been made to the content of the directive.

This window is a simple text editor. Activate the ‘NEW PARAGRAPHS AT CHARACTERS’ directive by deleting the text ‘COMMENT : ’ in front of it. N.B. Do not delete the ‘\*’ — this is a necessary part of the syntax of all DELTA directives. Next, delete all the numbers at the end of the directive, except ‘1–2’. The directive should now read

```
*NEW PARAGRAPHS AT CHARACTERS 1–2
```

This means that, in the descriptions, new paragraphs will be started at character 1 (flower colour), and character 2 (leaf length). This is hardly necessary now, with only 2 characters, but will be appropriate later when more flower and leaf characters are added. (The number ‘2’ in the directive will be changed automatically as more flower characters are added above the leaf characters.)

Similarly, activate the ‘ITEM SUBHEADINGS’ directive, and make it read

```
* ITEM SUBHEADINGS
#1. \b{}Flowers.\b0{}
#2. \b{}Leaves\b0{}
```

(Delete the rest of the subheadings, leaving only these two.) The window should now look like this.

```
Grevillea.dlt [3] (Copy of C:\delta\Grevillea\layout)
*COMMENT ~ Layout for natural-language descriptions.
*REPLACE ANGLE BRACKETS
*OMIT CHARACTER NUMBERS
*OMIT INNER COMMENTS
*OMIT INAPPLICABLES
*COMMENT Specify slot in which image links will appear in HTML descriptions.
There must be a corresponding character in the character list, and no data
should be recorded against this character in the item descriptions (the data
come from the TAXON IMAGES directive in the file 'timages').
*COMMENT : CHARACTER FOR TAXON IMAGES 88
*COMMENT : EXCLUDE CHARACTER 89
*NEW PARAGRAPHS AT CHARACTERS 1-2
*COMMENT Group characters into sentences.
*COMMENT : LINK CHARACTERS 3-5 7-9 10-11 13-15 16-17 18-21 26-29 32-37 41-42
45-47:52-54 48-51 55-56 57-59 62-63 64-65 68-70 71-72 73-75 78-84
*COMMENT Headings within in taxon descriptions. The numbers specify the
character before which the heading is placed.
*ITEM SUBHEADINGS
#1. \b{}Flowers.\b0{}
#2. \b{}Leaves.\b0{}
I
```

Save the changes by selecting ‘File > Save text’ in the main window (not shown above), and close the editing window. (Saving the file causes the syntax to be checked. If there are errors, you will have to correct them before the action set can be saved. See the [User’s guide to the DELTA System](#) for detailed information about the meaning and syntax of directives.) Then save the whole dataset (‘File > Save Dataset’).

Now run the ‘tonatr’ action set, and look at the output, as described in [Experimenting with the sample data](#).

Also try running all the other kinds of action set mentioned in [Experimenting with the sample data](#). All of the action sets will run without altering the templates, but to get good results you will usually need to change some of the directives.

Numeric characters (such as ‘leaf length’) cannot be used in conventional keys unless they are converted into multistate characters during the ‘Translate into KEY format. (tokey)’ run. Therefore you should activate the KEY STATES directive in this action set, and change it to read (for example)

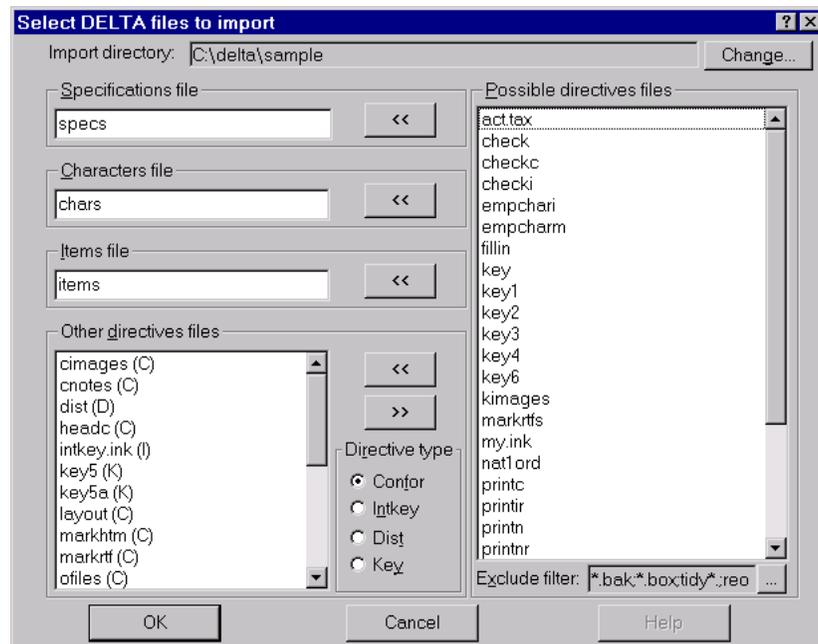
\*KEY STATES 2,~5/5-15/15~

which defines three states for 'leaf length': '1. up to 5 cm long/ 2. 5-15 cm long/ 3. 15 cm long or more/'.

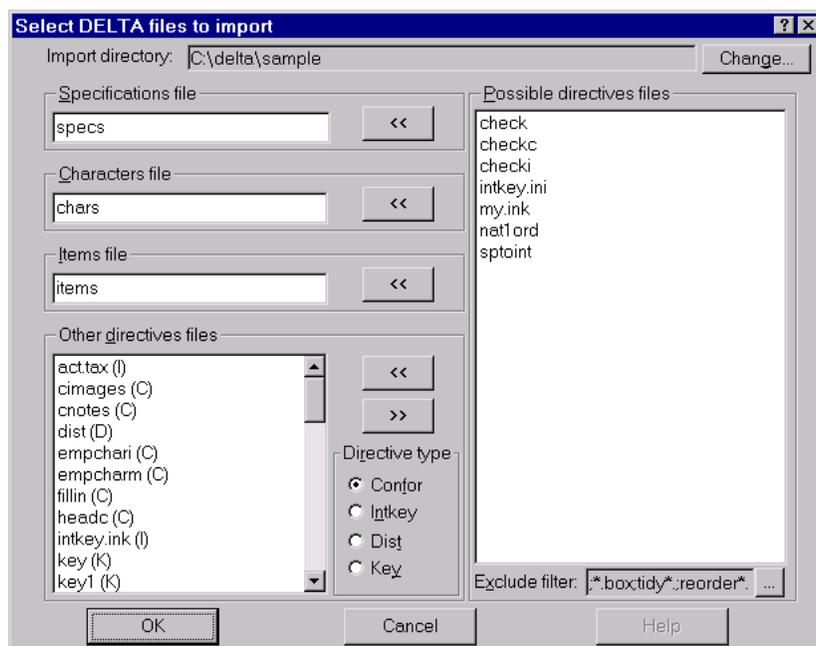
The action sets 'tonath' and 'tonatsr', which produce natural-language descriptions in separate files for each taxon, require a text character in which you must record the name of the output file for each taxon (without the file extension or type (.rtf or .htm), which is added by the program). These names must not be longer than 8 symbols, and alphabetic symbols should be in lower case.

## Importing DELTA text files

To import existing DELTA text files, first start a new dataset by selecting 'File > New Dataset'. Then select 'File > Import directives'. This will open a dialog to allow you to select the directory in which the DELTA files reside. (All directives files must reside in the same directory.) After choosing the directory, a new dialog will open and present you with a list of files in the folder that appear to be DELTA directives files. (The program examines the files to determine whether the first non-blank character in the file is '\*'.) The program assumes that the external files are split into the usual arrangement of 'specifications', 'characters', 'items', and other directives files. Here is the result of selecting the DELTA 'sample' directory.



Move the names of the files you wish to import to the boxes on the left of the dialog by selecting them in the box on the right and clicking the appropriate '<<<' button. Note that some names are already in the left-hand boxes — the names of the main data files ('specs', 'chars', and 'items'), and the names of the action-set templates supplied with the programs. You should set the appropriate 'Directive type' (Confor, Intkey, Dist, or Key), before or after moving the name(s). A letter in parentheses to the right of the name of each selected file indicates the 'Directive type' you have specified for that file.



Some files are not being imported for various reasons. (The ‘check’ files are not needed, as the Editor always maintains the correct syntax. ‘intkey.ini’ is the same as ‘intkey.ink’. ‘natord1’ is a Delfor file, which contains directives not recognized by the Editor.) Click ‘OK’ to begin reading the files. An ‘Import Status’ dialog shows progress, and a check box allows you to specify whether to ‘Pause on errors and messages’. Another window contains a log of the importing process. When all the files have been imported, click the ‘Done’ button to close the ‘Import Status’ dialog. You can then save, print, or close the log.

There are two special points to note about the format of the DELTA directives files. Firstly, they should use the ANSI, rather than OEM (also called ‘IBM’ or ‘ASCII’), encoding for accented characters. Secondly, if formatting information (e.g., italics, bold, etc.) is included, it should be in the form of RTF commands. However, the directives files should *not* be true RTF documents; they should only use RTF markup conventions. That is, they should not include a full RTF header, but should only make use of RTF markup commands, such as ‘\i{ }’ to initiate italics and ‘\i0{ }’ to end italics. (The use of ‘{ }’ instead of a space to terminate an RTF instruction is unusual, but allowed by the RTF specification. It is necessary to use this form for the time being, because the current Confor may remove spaces.)

For more information about RTF marks, see the [User’s guide to the DELTA System](#).

## Saving or exporting a dataset

A dataset is normally saved in a binary format. Unlike the ‘old’ DELTA files, all information about items, characters, dependencies, directives, and so on is saved in a single binary file. This file should normally be given the extension ‘.dlt’. When you make changes during a session, those changes are not made to the ‘dlt’ file until you explicitly save them. This is done via the ‘File > Save’ or ‘File > Save As’ options of the main menu. If you have made changes but not saved them, the program will ask whether you want to save or discard your changes when you close the document (or exit the program).

A dataset can also be exported into old-style, human-readable directives files, as described in the [User’s guide to the DELTA System](#). To do so, select ‘File > Export directives’ from the menu. This will open a dialog to allow you to select the directory in which the DELTA files should be placed. After choosing the directory, a new dialog will open, allowing you to specify the files to be exported. After specifying the files, Click ‘OK’. While the files are being written, a status dialog will show the progress of the procedure. Existing directives files in the target directory will be renamed to have the extension ‘.bak’.

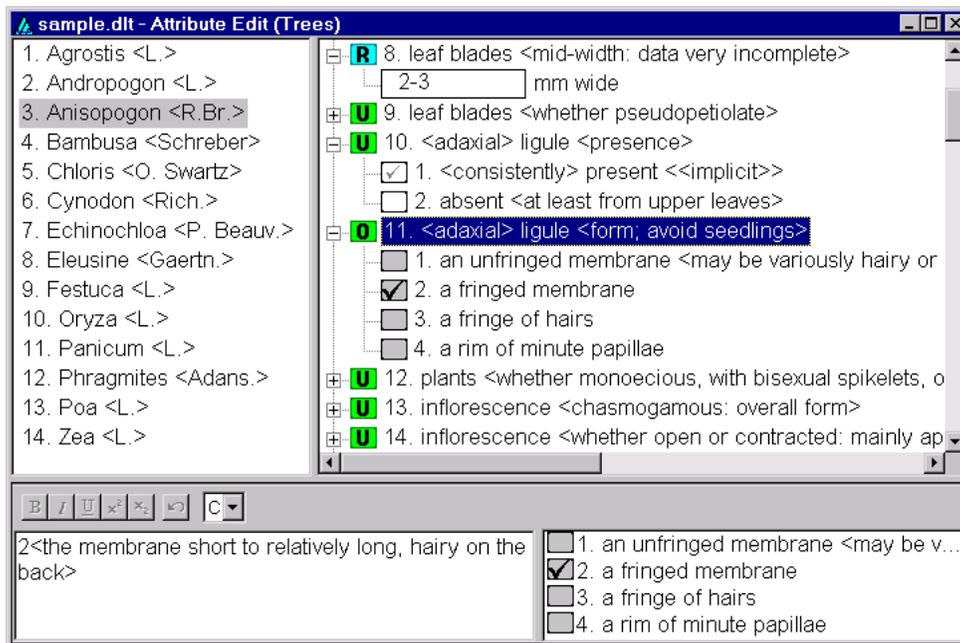
## Views of the data

There are several ways of viewing and editing the data. It is possible (and often convenient) to view your data in several different ways simultaneously. The default view is currently the 'tree-view' attribute editor. To open a new view of a dataset, select 'View', then select the required view.

## The attribute editors

### The tree-view attribute editor

This is the default means of viewing the data. Here is an example.



The window is divided into four panes, which can be resized by the user as desired by clicking on and dragging the boundaries between them.

*The Item Tree (top-left pane).* This contains a list of the names and numbers of all items (taxa) in the dataset. To select an item as the 'current' item for the window, click on it (i.e. on its name or number) with the left mouse button (item 3, Anisopogon, is the currently selected item in the example above). To move an item to a different position in the list, drag it with the left mouse button (that is, move the mouse while holding down the button). To copy (or move) an item, drag it with the right mouse button — a popup menu offers a choice between copying or moving. Clicking the right mouse button opens a popup menu with options for inserting, modifying or deleting items in the list. Double-clicking on an item will open the item editor dialog.

(The reader might wonder why this pane is called the item *tree* when it appears to be a simple *list* of items. In future versions of the program, it will be possible to group items hierarchically. When that is implemented, a tree structure will be used for navigating the hierarchy.)

*The Character Tree (top-right pane).* The upper-level nodes of this tree consist of a list of the feature descriptions of all characters in the dataset. An icon to the left of the feature description indicates the character type — whether **T**ext, **I**nteger numeric, **R**eal numeric, **U**nordered multistate, or **O**rdered multistate. When a node is expanded, it reveals either (for text and numeric characters, such as character 8 in the example above) a text box giving the value of that character for the selected item (and possibly the units of measurement); or (for multistate characters, such as characters 10 and 11 in the example above) a list of the states defined for that character.

To select a character as the 'current' character for the window, click on it (anywhere in its feature or state description) with the left mouse button (character 11 is the currently selected character in the example

above). To move a character to a different position in the list, drag its feature description with the left mouse button. To move a state within its character, drag the state description with the left mouse button. Clicking the right mouse button opens a popup menu with options for inserting, modifying or deleting characters and their states. Double-clicking on a character will open the character editor dialog.

For multistate characters, a check box appears to the left of each state description. If a tick is present in the box, it indicates that the state applies to the currently selected item. A smaller, grey tick indicates that the state is present by implication, rather than explicit coding (see character 10 in the example). If the background colour of the box is white, the application of that state to the currently selected item may be toggled by clicking on the box.

A white background in a character-value box, as in characters 8 and 10 in the example, indicates that the attribute is ‘simple’, and the value(s) can be edited *in situ*. A grey background, as in character 11 in the example, indicates that the attribute is ‘complex’, defined as follows.

- *Text characters*. The attribute has embedded formatting information.
- *Numeric characters*. The attribute includes comments, extreme values (enclosed in parentheses), or delimiters other than ‘-’ (‘to’).
- *Multistate characters*. The attribute includes comments, or delimiters other than ‘/’ (‘or’); or the states within the attribute are not in ascending order.

Complex attributes can be edited or entered only in the complex-attribute editing pane — see below.

If a character is inapplicable because of dependency relationships, a diagonal red bar appears across both the ‘type’ icon and the state check boxes.

*The Complex-Attribute Editor (lower-left pane)*. This pane is a rich edit control — that is, it allows display and editing of formatting marks. It contains the DELTA encoding for the currently selected item and character, using the traditional DELTA style. Angle brackets should appear around comments, but should *not* be placed around text attributes. A toolbar above the pane provides controls for rendering portions of the text in bold, italics, etc., and for undoing changes. Error checking is performed when the input focus leaves the edit control.

*The State List (bottom-right pane)*. If the currently selected character is a text or numeric character, this pane contains only an icon denoting the character type, and the units of measurement (if applicable). If the currently selected character is a multistate character, it contains a list of the states of the character, with a check box next to each. The appearance and method of use of the check boxes is the same as in the character-tree pane. This pane is provided so that the codings may be viewed and edited when the character is not ‘expanded’ in the character-tree pane.

## The grid-view attribute editor

This view is similar to that of a spreadsheet program. Here is an example.

		10	11	12	13	14
		<adaxial> ligule <presen...	<adaxial> ligule <form; avoid seedlings>	plants <whether monoecious with bisexual spikelets, or dioecious>	inflorescence <chasmogamous: overall form>	inflorescence <whether open or contracted: mainly applied to panicles>
1	Agrostis <L.>	1	1	2	5	1/2
2	Andropogon <L.>	1	1-2	2	2/5	
3	Anisopogon <R.Br.>	1	2	2	5	1
4	Bambusa <Schreber>	1	1-2	2		1
5	Chloris <O. Swartz>	1	2-3	2	2	-
6	Cynodon <Rich.>	1	2/3	2	2	-
7	Echinochloa <P. Beauv.>	1/2	3	2	2	
8	Eleusine <Gaertn.>	1	2	2	2	1/2

The upper pane of this view is a grid displaying the data matrix. The columns represent the characters of the dataset and the rows represent the items. The cells of the matrix display the attribute corresponding to a given character (column) and item (row). For non-text characters, any comments are removed from the attribute before display. This allows more of the 'basic' data to be displayed. When an attribute appears in grey text (as for many of the entries in the column for character 10 in the example), it indicates that the state shown is present implicitly, rather than by explicit encoding. A grey background, as for character 11 of most taxa in the example, indicates that the attribute is 'complex' (see definition above) and can be edited or entered only in the complex-attribute editing pane. A focus rectangle indicates the currently selected attribute of the grid. Elements of the pane may be resized by dragging their boundaries in the column or row headings (the grey areas at the left and top).

Characters and items may be re-ordered by dragging a feature name or taxon name (that is, the grid's column or row headings, respectively) to the desired position. Dragging with the right mouse button either moves or copies the item (for rows) or character (for columns) — a popup menu offers the choice 'Move here' or 'Copy here'. Clicking the right mouse button opens a popup menu with options for inserting, modifying or deleting items or characters. Double-clicking (with the left mouse button) on a character name will open the character editor dialog. Double-clicking on an item name will open the item editor dialog.

To copy cells to the clipboard, select the cells you want, and choose 'Edit > Copy' from the main menu. Note that the keyboard shortcut is 'Ctrl+Ins', not the standard Windows 'Ctrl+C'. The 'Paste' option has not been implemented, but the clipboard can be pasted into other programs such as Excel or Word. When pasting into Excel, start a new sheet, select the whole sheet, and format the cells as text (otherwise DELTA codes containing a slash will be interpreted as dates). Select the top left cell, and paste.

There are 'Print', 'Print Preview', and 'Print Setup' options in the 'File' menu (in grid view only). If you want more control over the layout, you can copy part of the grid and paste it into a word processor or spreadsheet. In Word, paste into a table, or paste as text and then use 'Table > Convert > Text to Table'.

The bottom portion of the window consists of the same panes for editing an attribute as were described for the Tree View window.

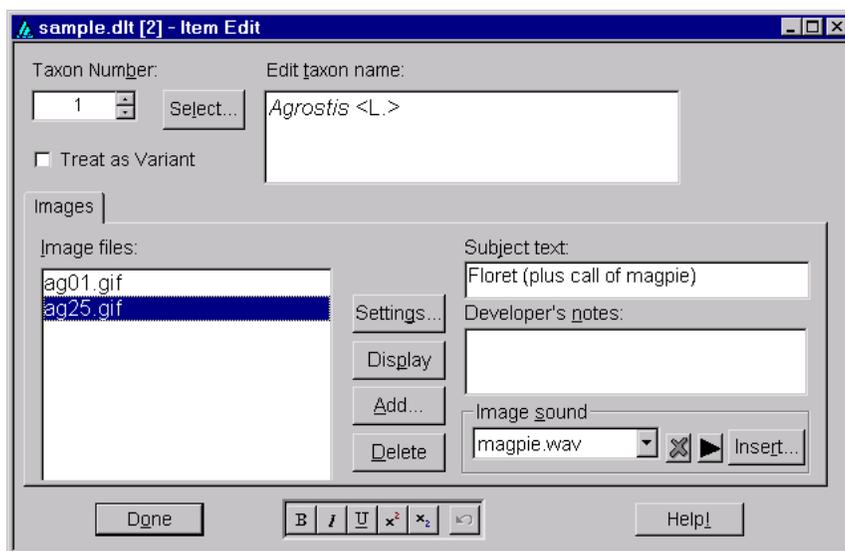
## Using the ‘Enter’ key in attribute editing

When editing attributes in either the tree or grid view, pressing the ‘Enter’ key causes a skip to the next taxon, or the next applicable character, and ‘Shift+Enter’ causes a backward skip. If the taxon-tree pane is active, skipping is by taxa, and if the character-tree pane is active, skipping is by characters. Otherwise, the type of skipping is determined by a drop-down listbox in the toolbar.

Clicking ‘Ctrl+Enter’ in the complex-attribute editing pane inserts a new-paragraph mark.

## The item editor

This dialog is provided for the creation or modification of new items and their names. It is opened either by choosing ‘View > Item editor’ from the menu; by right-clicking the mouse in the upper left pane (the Item Tree) of the Tree View window, and selecting ‘Insert new item at this position’ or ‘Append new item to end of list’ from the resulting menu; or by double-clicking on an item name in either the Tree or Grid View. The dialog appears as shown here.



In the upper-left portion of the dialog, a small up-down control allows selection of the taxon that is to be modified. Alternatively, the taxon can be selected from a list by clicking the ‘Select’ button. A rich edit control in the upper-right corner is used for entering or modifying the taxon name. A toolbar at the bottom of the window provides controls for rendering portions of the text in bold, italics, etc., and for undoing changes.

The ‘Images’ panel, which occupies the bottom half of the dialog, allows images to be associated with the taxon. On the left side of the panel is a list of the names of image files associated with the taxon. (Note that the program stores only the names of image files within the DELTA data file. It does not store copies of the images themselves). **N.B. Although the Editor will accept blanks in image file names, such names should be avoided, as they will not work with other DELTA programs, which use blanks as delimiters.** The names within this list may be re-ordered by dragging and dropping with the left mouse button. Clicking on the list with the right mouse button opens a popup menu with options for inserting, modifying or deleting images. Double-clicking (with the left mouse button) on an image file name will cause that image to be displayed in a window of its own.

**We recommend that the images be stored in a subdirectory ‘images’ of the main data directory, and that this directory be set as the ‘Image Path’ before linking any images.** To set the ‘Image Path’, click the ‘Settings...’ button, and enter ‘images’ (without the quotes) in the ‘Image Path’ box in the ‘Image Settings’ dialog. If an image path is not set in this way, the full path name is stored in the image link, and this can cause problems if the data are moved to another computer or published on the Web. The image path must also be set in the Intkey initialization file, intkey.ink (see ‘Generating an interactive

key', above). The default intkey.ink, produced when a new dataset is started, defines the image path as 'images', corresponding with the the path recommended above.

Clicking the 'Settings' button located near the centre of the panel will open the 'Image Settings' dialog (described below). Clicking the 'Display' button causes the currently selected image to be displayed. Clicking the 'Add' button allows you to preview and select a new image file to insert in the list (the image is inserted just above the current selection). Clicking the 'Delete' button caused the currently selected image file to be removed from the list. In the upper right portion of the panel is an edit control for entering a 'subject text' for the image. This text is used as a brief label for the image; it is used in the 'Subject' menu and caption of the displayed image to identify the image. A second edit control below this provides a space for entering developer's notes about the image.

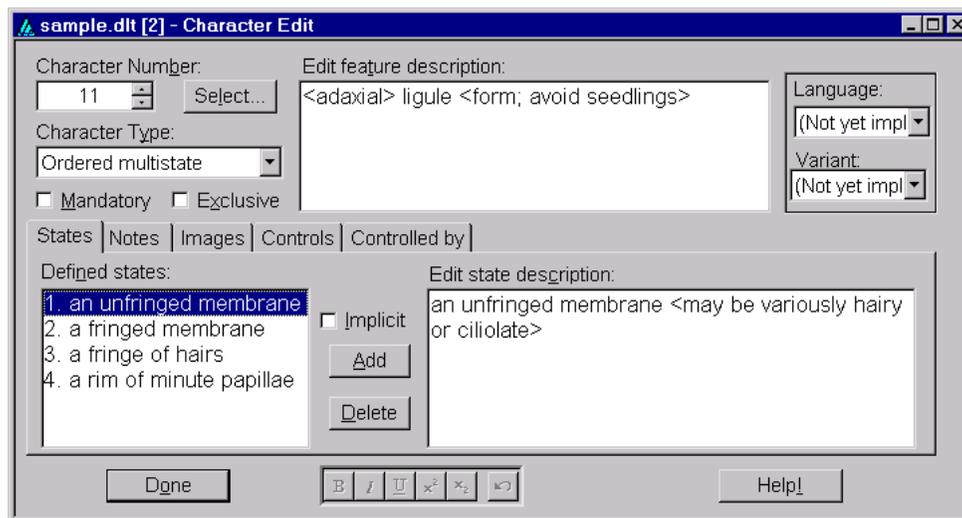
In the lower right portion of the panel are controls for associating and previewing one or more sound files with the current image. (As with images, only the names of sound files are stored within the DELTA file. The files themselves are not stored.) A drop-down list box in this region lists the sound files associated with the image. Clicking with the right mouse button on the list opens a popup menu with options for inserting, modifying or deleting sounds. Clicking the x-shaped button causes the currently selected sound to be deleted, and clicking the arrow-shaped button plays the sound. The 'Insert' button allows you to locate, preview, and select new sounds for the image.

The 'Done' button in the lower-left corner causes the dialog to close.

The 'Help' button in the lower-right corner is not functional at present.

## The character editor

This dialog is provided for the creation and modification of characters and their associated states. It is opened either by choosing 'View > Character editor' from the menu; by right-clicking the mouse in the upper right pane (the Character Tree) of the Tree View window, and selecting 'Insert new character at this position' or 'Append new character to end of list' from the resulting menu; or by double-clicking on a feature description in either the Tree or Grid View. Here is an example.



This is a relatively complex dialog, with a number of components.

*The character number selector.* At the upper-left of the dialog is a small up-down control showing the number of the character being edited. The value shown here can be changed to select a different character for editing. Alternatively, a character can be selected from the character list by clicking the 'Select' button.

*The character type selector.* Just below the character number selector is a drop-down listbox that indicates the character type. The type should be chosen for any newly define character. It is not possible

to freely reassign the type of a character once it has been defined, provided no items are encoded for that character, and no controlling attributes are defined by it.

*The mandatory character checkbox.* This indicates whether the character is ‘mandatory’, that is, must be recorded in every item.

*The exclusive character checkbox.* This checkbox, which appears only for multistate characters, indicates whether states of the character should be ‘exclusive’, that is, whether only a single state of the character can be associated with any given item.

*The feature description editor.* This is located at the top centre of the dialog, and is a rich edit control containing the ‘feature’, or basic description of the character. As with ‘conventional’ DELTA coding, the feature description should usually begin with a lower-case letter.

*The language selector.* This is at the upper-right of the dialog. It is intended to allow selection of alternative languages. This feature has not yet been implemented.

*The wording variant selector.* This is near the upper-right corner of the dialog. It is intended to allow the user to select among alternate wordings of the character list for different purposes (such as key generation vs. natural language descriptions). This feature has not yet been implemented.

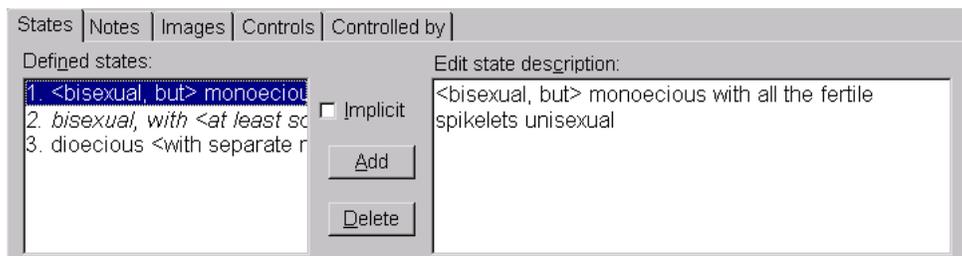
*The ‘Help’ button.* This is at the lower right of the dialog. It is currently not functional.

*The formatting toolbar.* This is located at the bottom centre of the dialog. It provides buttons for rendering text in bold, italics, etc., and for undoing editing operations. The toolbar will operate with whichever rich edit control is currently active.

*The ‘Done’ button.* This is at the lower left of the dialog. It closes the dialog.

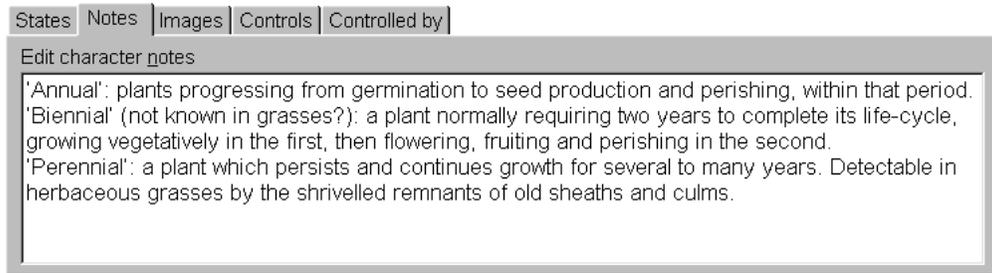
In mid-dialog are a series of different controls, each providing access to a different aspect of the character. These are selected via a tab mechanism.

## The state editor



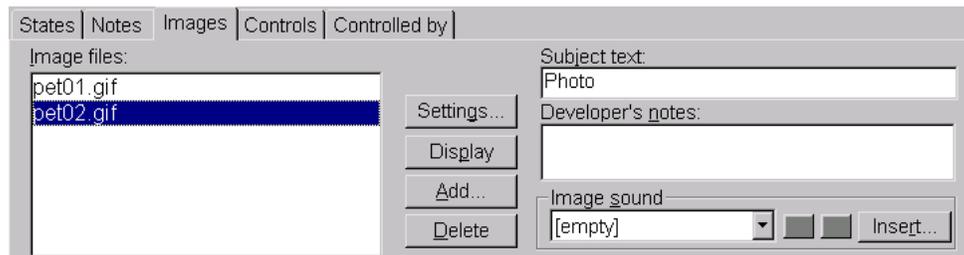
This panel contains a list of currently defined states (on the left), a rich edit control (on the right), and a check box and buttons (in the middle). The user selects among states via the list control on the left. The description of that state may then be edited in the rich editing control on the right. Clicking the ‘Add’ button indicates that a new state is to be inserted above the currently selected state. Clicking the ‘Delete’ button deletes the currently selected state. New states may be appended to the list by moving the list selection to the line below the last already-defined state, then entering the new state description in the state description editor. Pressing the ‘Enter’ key while in the description editor will cause the selection to move forward to the next state. This can be used to append a series of new states to the character without having to repeatedly use the list box and add button. States in the list control can be re-ordered via dragging-and-dropping. The ‘Implicit’ check box allows the specification of the currently selected state as the ‘implicit’ state. When the character is not recorded for a taxon, the implicit state value is assumed. The name of the implicit state (if one exists) will be shown in italics in the ‘Defined states’ list; state 2 in the figure above is implicit.

## The notes editor



This panel consists of a single, large, rich edit control. Supplementary information about the character may be entered here.

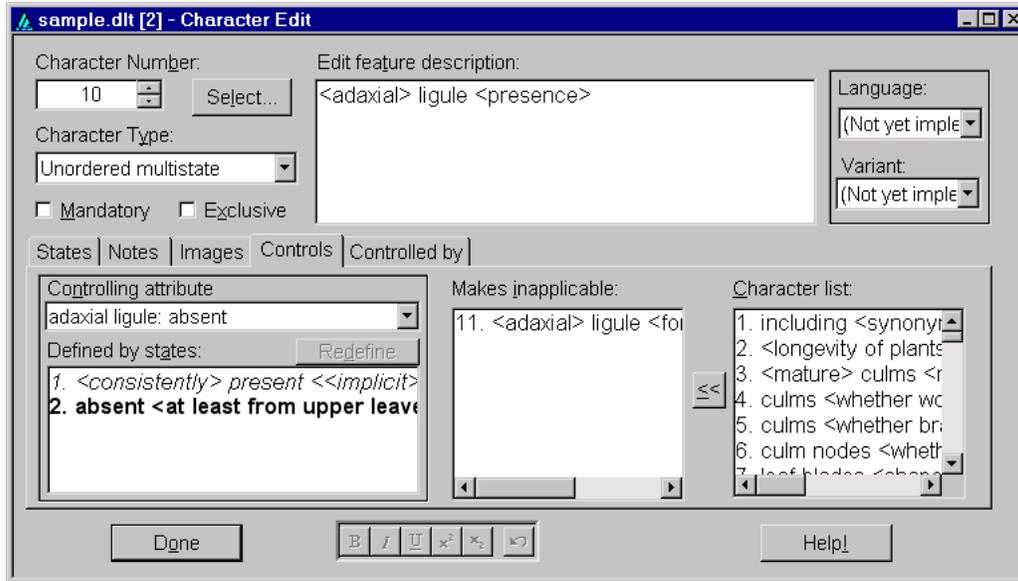
## The image-association editor



The 'Images' panel allows the association of images with the character. On the left side of the panel is a list of the names of image files associated with the character. (Note that the program stores only the names of image files within the DELTA data file. It does not store copies of the images themselves). **N.B. Although the Editor will accept blanks in image file names, such names should be avoided, as they will not work with other DELTA programs, which use blanks as delimiters.** The names within this list may be re-ordered by dragging and dropping with the left mouse button. Clicking on the list with the right mouse button opens a popup menu with options for inserting, modifying or deleting images. Double-clicking (with the left mouse button) on an image file name will cause that image to be displayed in a window of its own.

**We recommend that the images be stored in a subdirectory 'images' of the main data directory, and that this directory be set as the 'Image Path' before linking any images.** To set the 'Image Path', click the 'Settings...' button, and enter 'images' (without the quotes) in the 'Image Path' box in the 'Image Settings' dialog (described below). If an image path is not set in this way, the full path name is stored in the image link, and this can cause problems if the data are moved to another computer or published on the Web. The image path must also be set in the Intkey initialization file, intkey.ink (see 'Generating an interactive key', above). The default intkey.ink, produced when a new dataset is started, defines the image path as 'images', corresponding with the the path recommended above.

## The controlling-attribute editor



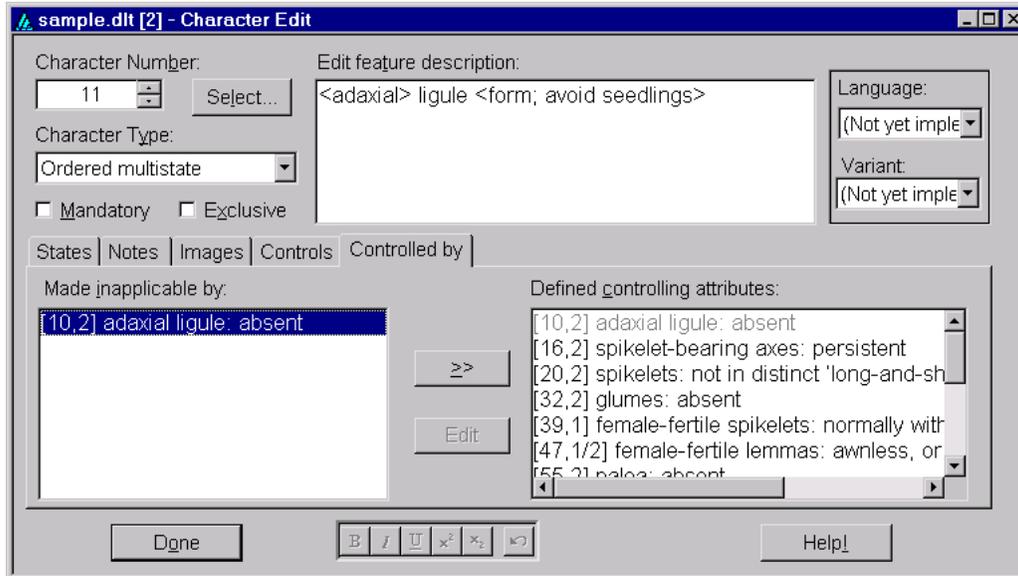
This panel is used for defining ‘controlling attributes’; that is, combinations of one or more states within a character, which when encoded for in a given item can make other ‘dependent’ characters inapplicable. It can also be used to define the associations between the controlling attributes and the corresponding dependent characters.

The example above is associated with a character that encoded ligules as being present or absent. A controlling attribute (‘ligule: absent’) is defined here, and is used to make inapplicable another character dealing with ligule shape.

The panel contains a drop-down list of controlling attributes defined for this character in the upper left corner, a list box of the character’s states in the lower-left corner, and two lists to the right containing characters within the dataset. To define a new controlling attribute, select ‘[new]’ from the list of controlling attributes. Then use the state list box to select those states that should constitute the new controlling attribute (the selected states will appear in a bold font). When all relevant states have been selected, click the ‘Define’ button to define the new controlling attribute. You will be given an opportunity to assign an identifying label to the controlling attribute; by default, a label will be generated from the character’s feature and state descriptions. (However, note that there is currently no provision for exporting custom labels to DELTA format.)

Characters are placed under (or removed from) the control of a given controlling attribute by use of the two list boxes on the right side of the panel. The list on the far right includes all characters in the dataset. They may be added to the ‘inapplicable’ character list either by selecting them and clicking the move (‘<<’) button, or by dragging-and-dropping to the other list. A similar procedure can be used to remove characters from the ‘inapplicable’ list.

## The 'Controlled-by' chooser



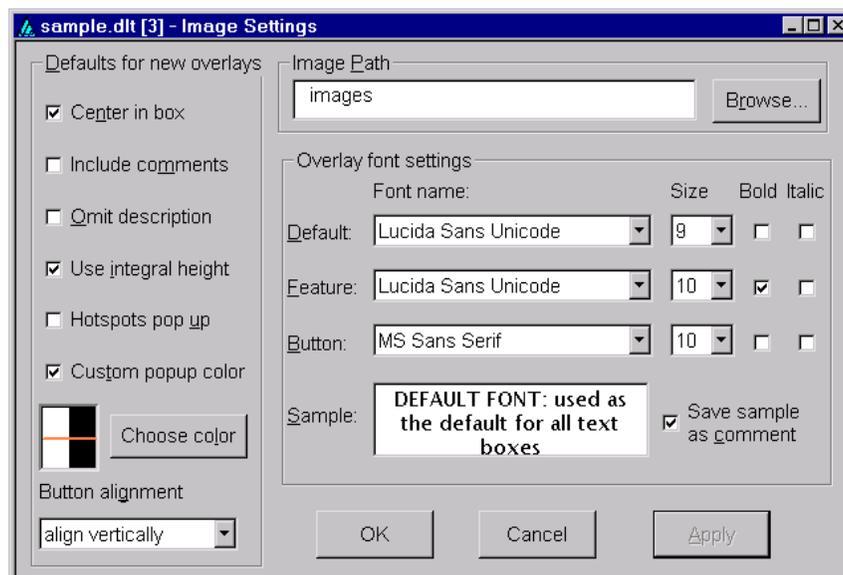
This panel provides an alternative way of associating dependent characters with their controlling attributes. Note, however, that controlling attributes cannot be *defined* in this panel.

The panel shows controlling attributes that control the current character. On the right of the panel is a list of all controlling attributes defined for the dataset. On the left is a list showing those that make the current character inapplicable. Controlling attributes may be moved to (or from) the 'made inapplicable by' list either by selecting them and clicking the 'Move' button, or by dragging-and-dropping them.

The 'Edit' button is not yet functional, but is intended to take the user to a view where a given controlling attribute was defined.

## The image settings dialog

This dialog is provided to change settings related to display of images and their text overlays. It is opened either by choosing 'View > Image settings' from the menu; by clicking on the 'Settings' button of the images panel of the Item Editor or Character Editor windows; or by right-clicking the mouse on an image file name in an images panel or a displayed image and selecting 'Image settings' from the resulting menu. Here is an example.



The left portion of this dialog selects default characteristics to be applied to newly created text overlays on images. Changing these settings generally has no effect on existing overlays. An exception is that changing 'Button alignment' will affect existing push-button overlays if those overlays are later edited or repositioned on their image. For more detailed information on the meaning of these settings, see the section below on Image Windows and Overlays.

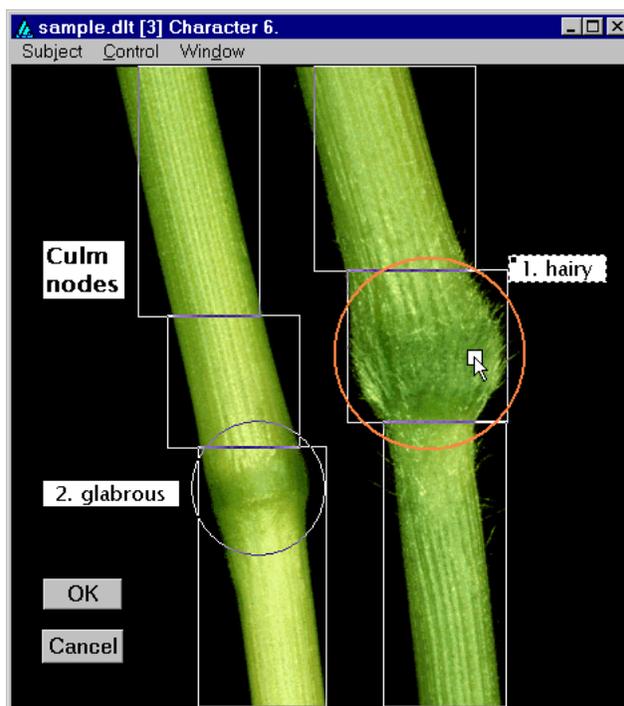
An edit control in the upper right portion of this dialog contains the 'image path'. This is a list of directories in which the program will look for image or sound files. Multiple directories are separated with a semicolon. For example, an image path of '.;images;e:\' will cause the program to search for images in the current directory (that is, the directory in which the '.dlt' file itself resides), the subdirectory 'images' of the current directory, and the root directory of the E: drive. Components of the image path may be URL's, e.g., <http://delta-intkey.com/angio/images>. The 'Browse' button can be used to locate directories on the local machine to be added to the image path. URL's must be entered manually.

The lower left portion of the dialog provides a mechanism for setting the fonts to be used for image overlays. The 'default' font is used for most text overlays. The 'feature' font is used for the display of feature description overlays; if it is not set explicitly, the 'default' font is used. The 'button' font is used for the 'OK', 'Cancel', and 'Notes' buttons. Changes to these values will affect all existing overlays, as well as those yet to be created. A sample of each font is displayed as changes are made.

Note that changes made within this dialog do not take effect immediately. You must click either the 'Apply' button or 'OK' button (which also closes the dialog) for any changes to be applied.

## Image windows and overlays

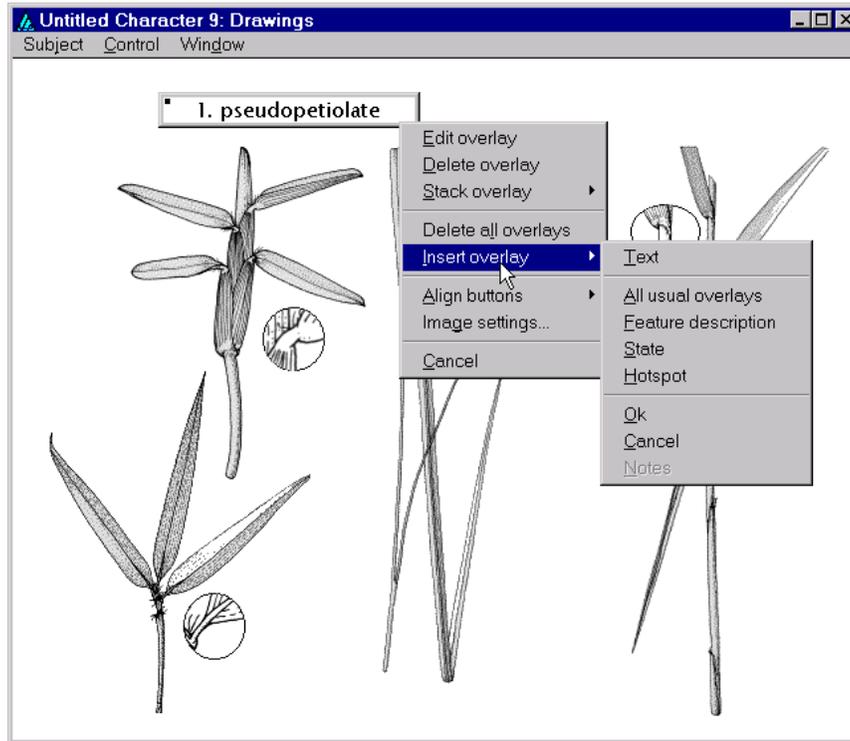
Character and taxon images are displayed in windows that are meant to closely resemble the appearance they have within Intkey. These images may have *overlays* superimposed upon them. The editor provides facilities for adding, modifying, and deleting these overlays. Here is an example of a character image, as displayed in the editor.



Note that the image window contains its own menu. The 'Subject' menu item allows the user to go to a different image attached to the same taxon or character, based on the subject text of the image. The 'Control' menu provides options for going on to the next or previous image associated with the current taxon or character, or for going on to the next taxon or character. The 'Window' menu provides options

for controlling how the image and its overlays are displayed, and for obtaining information about the image.

The image above contains examples of the basic different types of overlays: text boxes (which automatically become scrollable for large amounts of text), push buttons, and 'hotspots'. Overlays are inserted, modified, or deleted by right clicking on the image or an existing overlay. This will cause a popup-menu to appear, as seen below.



The range of overlay types that are available depends upon the type of image being viewed. All image types allow for the creation of 'OK' and 'Cancel' buttons, which can be used in Intkey to close the image window; and for the creation of general-purpose 'text' overlays, which may contain any arbitrary text. Taxon images may also contain a text overlay to contain the name of the taxon, and a 'Notes' button which, when clicked, will cause notes associated with the image to be displayed in a separate window. Similarly, character images may contain a 'feature' text overlay, which displays the feature description of the character; and a 'Notes' button which, when clicked, will cause the character notes associated with the character to be displayed.

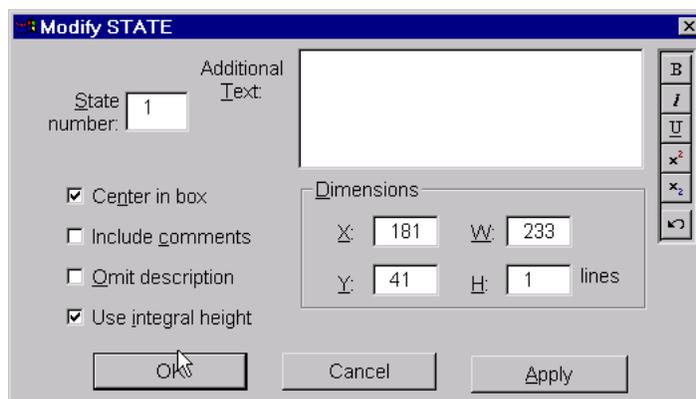
Character images may contain several other types of overlay, depending on the character type. Images associated with multistate characters may contain 'state' text overlays, which displays a state number and description. When used in Intkey, these overlays may be selected by the user to indicate the applicability of that state. The 'state' overlays may also have one or more 'hotspots' associated with them — regions of the underlying image that may also be used by an Intkey user to select the associated state. Similarly, images associated with numeric characters may contain selectable 'value' overlays that contain a numeric value (or range of numeric values), and these may also have associated hotspots. Images for numeric characters may, in addition, contain an 'enter' overlay in which an Intkey end-user may enter any value, and a 'units' overlay which displays the character's units of measurement (if they have been defined).

The basic overlays for a character image can be inserted in a single step by selecting the option 'All usual overlays'. This inserts the 'Feature description' box, 'State' boxes (for a multistate character) or 'Enter' and 'Units' boxes (for a numeric character), 'OK' and 'Cancel' buttons, a 'Notes' button (if there are notes for the character), and a hotspot for each character state.

Once they have been created, overlays may be resized or repositioned by use of the mouse (or similar pointing device). Click on a overlay to select it. Its border will become thicker. (The innermost margin of this border corresponds to the border of the 'normal' overlay window.) When the cursor is positioned

over this border, it changes to a double-headed arrow, indicating that the overlay window may be re-sized by holding down the left mouse button and moving the mouse. (The ‘OK’, ‘Cancel’, and ‘Notes’ push-buttons are exceptions to this — they may be moved but not resized.) To move the selected overlay window, but retain its size, position the cursor in the middle of the window and hold down the left mouse button. The cursor will change to a ‘hand’, and moving the cursor will move the window to a new position. An overlay may not be positioned beyond the boundaries of the image window. A ‘selected’ overlay may be returned to its normal state by ‘selecting’ another overlay, or by clicking on some portion of the image where there is no overlay.

Double-clicking the left mouse button within a ‘selected’ overlay will open a dialog box for modification of the properties of the overlay. The dialog may also be opened by right-clicking on the overlay and selecting ‘Edit overlay’ from the popup menu. A sample of such a dialog for a ‘state’ overlay appears as show here.



The four checkboxes along the left side of the dialog control aspects of how the text within the overlay will be displayed. A tick in the first checkbox indicates that the text should be centred (both vertically and horizontally) within its border. The default behaviour is for text to begin in the upper-left corner of its containing frame. Ticking the second checkbox indicates that any comments within the state description text should be displayed, with the surrounding angle brackets converted into parentheses. By default, comments are removed from the text. A tick in the third checkbox indicates whether the state description should be omitted entirely; the default behaviour is to display the state description text. The fourth checkbox indicates that the height of the frame containing the text should automatically be adjusted to contain an integral number of lines. By default, the text boxes may be of any height.

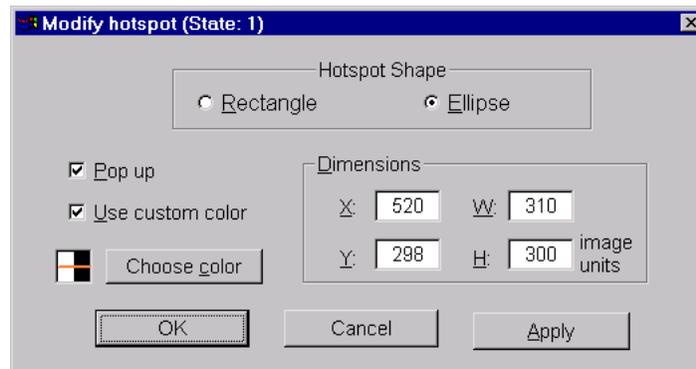
Any text entered in the ‘Additional Text’ area of the dialog will be appended to the state description text. If the state description text is omitted entirely, the text entered here will be displayed instead.

In the central-right portion of the dialog are controls for viewing and modifying the position of the upper-left corner of the overlay (‘X’ and ‘Y’) and the overlay’s width and height (‘W’ and ‘H’). These values are expressed in *image units*, not pixels; these are values in the range 0 to 1000, with a value of 1000 corresponding to the width or height of the entire image. However, if the ‘Use Integral Height’ checkbox is ticked, the height value (‘H’) is given in terms of lines of text. The upper-left corner of the image is always taken as the origin (0,0), with positive values to the right and downwards.

Note that changes made within this dialog do not take effect immediately. You must click either the ‘Apply’ button or ‘OK’ button (which also closes the dialog) for any changes to be applied.

The dialogs for modifying other types of text overlays are quite similar to the ‘state’ example shown here, and differ only in minor details. In the ‘Modify UNITS’ overlay (for numeric characters), the ‘X’ and ‘Y’ values may be set to ‘~’, denoting a position contiguous with the ‘Enter’ overlay. The dialogs for ‘button’ overlays allow only the button’s position to be modified.

A slightly different dialog is available for editing the properties of hotspots. Here is an example.



At the top of this dialog, a pair of radio buttons is used to select whether the shape of the hotspot is rectangular or elliptical. To the right centre are controls for viewing and modifying the position and extent of the hotspot, just as for text overlays. To the left are controls that affect how the hotspot boundaries appear. By default, the hotspots are invisible to Intkey user, with only a change in cursor shape notifying them when the mouse passes over a hotspot. If the 'Pop up' option is selected, the outline of the hotspot will become visible whenever the mouse passes over the hotspot or over any other hotspot or 'state' overlay associated with the same state as the hotspot. The 'Use custom color' option allows the designer to select the colour that should be used in drawing the outline of a pop-up hotspot. By default, the boundary lines are drawn by 'inverting' the colours of the display. Note that 'inverting' the colours a second time restores the original colours. Hence when hot-spot boundary lines drawn in this way exactly overlie one another, they disappear from view. If the use of a custom colour is selected, the colour to be used may be chosen from the dialog that appears when the 'Choose color' button is clicked.

## Entering special symbols

The Rich Edit controls used in the Delta Editor support a variety of special symbols and international characters, which can be entered via the Windows 'Character Map' utility. (If this isn't already installed on your system, you can install it as follows. (1) Click the Start button, point to Settings, and then click Control Panel. (2) Double-click the Add/Remove Programs icon. (3) Click the Windows Setup tab. (4) Double click Accessories. (5) Check Character Map, and then click OK. (6) Click OK.) In addition, the Delta editor provides a number of shortcut keys to ease entry of many of these characters. For the most part, these shortcuts are the same as the defaults provided in Word 97, but several additional shortcuts of particular use with biological data (e.g., ♂, ♀, ×, and the Greek alphabet) are provided. Here is a complete list.

Shortcut Keystroke	Resulting symbol
Ctrl + Num-	– (en dash)
Ctrl + Shift + space	(non-breaking space)
Ctrl + -	(optional hyphen – see Notes)
Ctrl + Alt + c	©
Ctrl + Alt + r	®
Ctrl + Alt + t	™
Ctrl + Alt + m	♂
Ctrl + Alt + f	♀
Ctrl + Alt + x	×
Ctrl + Alt + o	°
Ctrl + Alt + !	¡
Ctrl + Alt + ?	¿
Ctrl + Alt + +	±

Shortcut Keystroke	Resulting symbol
Ctrl + Alt + .	...
Ctrl + Alt + Num-	— (em dash)
Ctrl + ' followed by:	(acute accents)
one of A,a,E,e,I,i,O,o,U,u,Y,y	Á,á,É,é,Í,í,Ó,ó,Ú,ú,Ý,ý
one of C,c,D,d,L,l,N,n,R,r,S,s,T,t,Z,z	Ć,ć,Đ,đ,Ł,ł,Ń,ń,Ŕ,ŕ,Ś,ś,Ţ,ţ,Ž,ž
space	' (prime)
'	' (closing single quote)
"	” (closing double quote)
Ctrl + ` followed by:	(grave accents)
one of A,a,E,e,I,i,O,o,U,u	À,à,È,è,Ì,ì,Ò,ò,Ù,ù
`	‘ (opening single quote)
"	“ (opening double quote)
<	«
>	»
Ctrl + ^ followed by:	(circumflex)
one of A,a,E,e,I,i,O,o,U,u	Â,â,Ê,ê,Î,î,Ô,ô,Û,û
Ctrl + Alt + ^ followed by:	(hacek)
one of C,c,D,d,E,e,L,l,N,n,R,r,S,s,T,t,Z,z	Č,č,Ď,ď,Ě,ě,Ľ,ľ,Ň,ň,Ř,ř,Š,š,Ť,ť,Ž,ž
Ctrl + : followed by:	(umlaut)
one of A,a,E,e,I,i,O,o,U,u,Y,y	Ä,ä,Ë,ë,Ï,ï,Ö,ö,Û,Û,ÿ
Ctrl + ~ followed by:	(tilde)
one of A,a,N,n,O,o	Ã,ã,Ñ,ñ,Õ,õ
Ctrl + @ followed by:	(ring)
one of A,a,U,u	Å,å,Û,û
Ctrl + , followed by:	(cedilla)
one of C,c,K,k,L,l,N,n,R,r,S,s,T,t	Ç,ç,Ķ,ķ,Ļ,ļ,Ņ,ņ,Ŗ,ŗ,Ş,ş,Ţ,ţ
Ctrl + & followed by:	(ligatures)
one of A,a,O,o,s	Æ,æ,Œ,œ,ß
Ctrl + / followed by:	(slash)
one of L,l,O,o	Ł,ł,Ø,ø
c	¢
Ctrl + " followed by:	(double acute)
one of O,o,U,u	Ő,ő,Ű,ű
space	” (double prime)
Ctrl + g followed by:	(Greek)
one of A,B,C,D,E,F,G,H,I,J,K,L	Α,Β,Ψ,Δ,Ε,Φ,Γ,Η,Ι,Ξ,Κ,Λ
one of M,N,O,P,R,S,T,U,V,W,X,Y,Z	Μ,Ν,Ο,Π,Ρ,Σ,Τ,Θ,Ω,Ξ,Χ,Υ,Ζ
one of a,b,c,d,e,f,g,h,i,j,k,l	α,β,ψ,δ,ε,φ,γ,η,ι,ξ,κ,λ
one of m,n,o,p,r,s,t,u,v,w,x,y,z	μ,ν,ο,π,ρ,σ,τ,θ,ω,ς,χ,υ,ζ

## Notes

At present, the rich edit controls within the Delta Editor use the Arial font, and only those characters available in that font will be displayed properly. Earlier versions of the font may not be able to display all the characters listed above.

Future versions of the Editor may allow the user to select the font used in rich edit controls. When that occurs, the user should always attempt to select a font that can properly display all the symbols used within the dataset.

The key combinations outlined above are case-sensitive. On some keyboards, for example, the ‘:’ character is obtained by holding down the Shift key while pressing the ‘;’ key. Hence the prefix for placing umlauts above characters requires holding down the Shift and Ctrl keys simultaneously, then pressing the ‘;’ key. The most complex such operation is the prefix for placing haceks above characters – on some keyboards this requires simultaneously holding down the Shift, Ctrl, and Alt keys, then pressing the ‘6’ key. For the most part, these key combinations are intended to aid the entry of an occasional international character or symbol. If an entire dataset is to be entered in a non-English language, then it may be more convenient to install a suitable keyboard configuration for that language.

Many of the key sequences listed above involve a control key combination, followed by some other key. The state of the control key is ignored when processing the concluding key. For example, to generate the ‘é’ character, the ‘Ctrl +’ prefix may be followed either by ‘e’ alone or by ‘Ctrl +e’.

The complete set of characters shown above is available in any of the rich edit controls of the editor (that is, in any context where bold, italics, etc. can also be displayed). A subset of these characters – those that are also members of the ANSI Windows character set — are also available when editing directives files.

The rich edit controls do not display optional hyphens. However, they will be exported correctly.

Unlike the shortcut keys available in Word, these key combinations are not user-definable. If you think that additional symbols should be added to those listed above, contact the Delta development team.

## Planned enhancements and known bugs

### Editing the main data

Further consistency checking needs to be added in many areas.

‘Mandatory’ characters are not enforced, except when data are being imported.

When deleting characters, states, or controlling attributes, no feedback is currently presented to the user to indicate whether these objects are already in use.

Better feedback needs to be provided to the user when a newly encoded attribute renders previously coded attributes inapplicable. (Currently, the controlling attribute in the tree view is flagged with an exclamation mark in a red triangle, and the dependent attributes in the grid view are displayed with both the previously coded values and a red line indicating ‘not applicable’.)

Code for validating the changed contents of an attribute needs to be improved.

No user-interface mechanism is provided for changing the font face and size used by some of the controls.

Support for alternate wording of character sets (including multiple languages) has not yet been implemented.

### Editing other directives

Currently, all the directives normally contained in the files ‘specs’, ‘chars’, ‘items’, ‘cnotes’, ‘cimages’, and ‘timages’ can be edited via a Windows interface that ‘understands’ this information. All others must still be edited in the old way — as text. Proper interfaces will be provided for editing these other directives. In the meantime, the files containing them should be imported into the editor (and exported as

necessary) if characters, states, or taxa are added or moved. The editor keeps all directives consistent with these changes. Thus, the program Delfor is now unnecessary for most purposes.

The KEY CHARACTER LIST directive is not fully supported: it is not maintained when characters or states are added, deleted, or reordered.

## Producing output

All the functions of Confor will be incorporated into the Editor.

## Bugs

A list of known bugs is available at [delta-intkey.com/www/bugs.htm](http://delta-intkey.com/www/bugs.htm).

# Appendix 1. Hints on organization and maintenance of data

## L. Watson

This is an edited version of Section 12 of *A primer for the DELTA System* (Partridge, Dallwitz and Watson 1993).

The DELTA system is not merely a device for facilitating the application of computer technology to the organization and manipulation of taxonomic data. Its use should have a constructive, critical influence on fundamental aspects of taxonomic practice. It encourages conscientious acquisition of properly comparative data (essential for rational classificatory argument), and it ensures that due attention is paid to questions of character and character-state definitions. Therefore, if you intend to prepare automated taxonomic descriptions, it is best where possible to take full advantage by using the DELTA system from the start, rather than trying to automate data already on hand.

The following hints on preparing character lists, defining character states, encoding data, and file management reflect wide experience with practical applications of DELTA and associated programs. They are directed towards minimizing encoding errors, maximizing the information content of files, saving labour, and achieving stylistically acceptable descriptions and keys.

## Preparing character lists

Your character list is the first file you will need to prepare. Its quality (i.e., the taxonomic wisdom it embodies and the skill with which it is worded) will define the standard of the rest of the work.

(However, see also ‘Developing a character list: reproducibility’, *above*. Unless you are experienced in developing descriptive databases, it is advisable start by entering only a few characters, recording data for a few taxa, then testing all the applications that you intend to use. MJD.)

Make sure that all the states of a character are logically related and mutually exclusive. The following are examples of kinds of ‘character’ definition to be avoided.

- #102. leaves/
  1. green/
  2. yellow/
  3. wide/
  4. narrow/
- #103. leaves/
  1. narrow/
  2. yellow/
  3. serrate/
  4. shiny/
- #104. leaves/
  1. green/
  2. light green/

102 should be 2 characters ('colour of leaves' and 'width of leaves'), and 103 should be 4 characters. The states of 104 should be 'dark green' and 'light green'. A useful test is to ask yourself whether you would be happy to choose between the states as alternatives in a key. However, you must ask this question in relation to an arbitrary specimen: alternatives like those in 103 *can* work in a key when only a particular set of taxa is still to be separated, but they are still unsuitable for combining in a character definition. For detailed examples, see [Appendix 2](#).

Take special care over the initial construction of the character list, with a view to minimizing the need for subsequent changes. In particular, include from the start all the likely characters you can think of, and try to order them from the beginning in a fashion which you consider logical, in the hope of avoiding subsequent major alterations in their relative positions.

Try to present states within characters in sequences that are logical, at least to your own mind; and be consistent, in this respect, from one character to another. For example, arrange things in ascending order of size, number or complexity, or in a sequence which has some significance for you. In expressing presence/absence, always place the positive state first: placing the negative state first can lead to (conventional) keys with odd-looking couplets.

It is sometimes necessary to allow the likely requirements of key-making to influence the form of the character list. However, any early temptation to overdo this, to the detriment of the quality of your descriptions, should be strongly resisted. In particular, try to avoid using inclusive character states, such as

1. leaves glabrous/
2. leaves hirsute, hispid, or hoary/

They may be acceptable in keys, but their presence in your character list will lead to embarrassing 'lies' in some of your descriptions (hirsute leaves, for example, being rendered as 'hirsute, hispid, or hoary'). '2. leaves not glabrous <hirsute, hispid, or hoary>/' would be preferable, but still unsatisfactory. If this sort of thing is unavoidable in the early stages through shortage of precise information, include a properly broken down version of the character as well, and record it as best you can from the beginning. The inclusive one may then be jettisoned later, as the data improve; or if it is useful in key-making, it can be retained for that purpose but excluded when making descriptions.

Perusal of the examples in the [User's guide to the DELTA System](#) will indicate various ways in which imaginative use of the <comment> facility permits the development of a character list which is comprehensible in itself, and which at the same time gives presentable natural-language descriptions. Make generous use of comments and character notes (see the sample file 'cnotes') throughout your character list, to qualify character and state definitions, to cover questions of applicability, to give references, etc. Consider constructing the character list and notes as a detailed glossary, with references to relevant articles, illustrations, etc.

Many users will be dissatisfied with the results of using the same character list for preparing both natural-language descriptions and conventional keys. The latter require that every character be set out in full, whereas this is unacceptably clumsy for descriptions, where the character sequence provides context. The only solution is to maintain a second version of the character list, appropriately modified for key making. For example, in the sample characters file (see also [User's guide to the DELTA System](#)), a sequence of characters, starting from character 26, refers to the female-fertile spikelets. In most of these characters, the words 'female-fertile' are within angle brackets. These words are not necessary in the natural-language descriptions, because the characters appear under the subheading 'Female-fertile spikelets'. However, in keys, the omission of these words is often misleading. A separate, key-making characters file is needed, in which such words are outside the angle brackets. The directives file 'tokey' should reference this file instead of 'chars'.

## Formatting

If it is intended that descriptions and keys will ultimately be automatically typeset, the formatting information required for italics, etc. should be inserted in all character lists and data from the start.

## Choosing items

Intkey allows you to coalesce descriptions; for example, to prepare descriptions of genera from those of subgenera or species. Thus, if there is doubt or controversy about group delimitation, it is feasible to define some items at a lower taxonomic level than that generally being aimed at in the treatment. In addition to yielding descriptions serviceable for checking on the usefulness of alternative circumscriptions, this approach more readily accommodates future changes of mind, and may make your data more acceptable to a wider audience.

## Encoding of data: technical aspects

Use text characters and comments to qualify and/or amplify the data. Apart from adding necessary detail and desirable polish to descriptions, these may eventually be converted into new characters, and their presence from the beginning will help minimize the inevitable but exasperating job of backtracking through specimens, slides and literature. If artfully framed, comments can help ‘humanize’ the printed descriptions. Bear in mind, though, that comments and text characters are inaccessible to many programs. Confine use of text characters to situations where data cannot be satisfactorily represented in other types of character (synonyms, references, lists of specimens seen, etc.); and to maximize their accessibility via Intkey, try to standardize the style in which each is expressed in the descriptions. In deciding on the wording of a comment, perform the mental exercises necessary to determine how it will read in the finished product, not only *per se*, but in context with the surrounding data and comments.

DELTA is very versatile, and the same piece of information can often be expressed in several different ways. However, in the long run, some solutions will be preferable to others, depending on the uses to which the data will be put. In particular, if key-making has high priority, care should be taken to ensure that information of key-making value is not lost unnecessarily by inappropriate encoding.

For example, working at generic level with plants, one finds that as more information is gathered, more and more of the better known characters become ‘variable’ within genera. If these are habitually coded simply as variable, it becomes increasingly difficult to make acceptable keys: they become dependent upon obscure characters, which may well be ‘invariable’ merely through ignorance. So, instead of

105,1/2

consider

105<nearly always>,1<absent in *X. vulgaris*>

or

105,1/2<rarely>

Try to be consistent in applying comments such as <rarely>: there will be the option later of deleting all such records for key-making purposes, should this seem desirable. When dealing with quantitative characters (IN or RN), values outside the commonly observed range should be enclosed in parentheses, for example, 41,(1–)2–10(–15). Then the extreme values can be omitted when making keys, by means of the Confor directive USE NORMAL VALUES.

Most users will choose, when making natural-language descriptions, to OMIT INAPPLICABLES. In this case, the wording will often be much improved by this kind of device:

39,1/2 40<when present>,2

translating to

Spikelets with female-fertile florets only, or with incomplete florets. The incomplete florets when present distal to the female-fertile florets.

Future programs for key making, interactive identification, and calculation of phenetic distances will be capable of using information on the relative frequencies of character states within a taxon. It is worthwhile anticipating this by incorporating this information into your comments wherever possible. Use a consistent format for such comments, to facilitate retrieval or conversion of the information.

Make sure that your DEPENDENT CHARACTERS directive is as comprehensive as possible. This will allow the DELTA Editor and Confor to detect certain kinds of coding error, and allow Key the greatest scope in using characters which are sometimes inapplicable.

## Encoding of data — organizational aspects

Individual preferences and circumstances will largely decide the procedures by which data are acquired, transcribed, encoded and transferred to a computer file. Nevertheless, some generalisations are possible.

Any interruption resulting in a loss of concentration while transcribing, encoding or entering data into files is a major hazard. Refreshments breaks, for example, and especially 'liquid lunches', might best be postponed or foregone while such operations are in progress.

Obtain printouts of all newly entered or edited descriptions, and check them scrupulously against the original handwritten versions. For all but the most trivial of editing operations, this is a two-person job. The computer printout should be read aloud, to be checked against the original, handwritten version by somebody other than the person who typed it into the file. If all is going well, so that errors are few and far between, it may be necessary to test wakefulness by deliberately misreading from time to time! Some people (not necessarily from lack of motivation) seem unable to organize themselves and concentrate at the level necessary to achieve the essential standards of accuracy. They should be kept at safe distances from all aspects of data banking. When things are working smoothly, transcriptional and encoding errors should be sufficiently rare to permit conducting enquiries into individual cases, with a view to avoiding similar problems in future.

Use Confor to check all editing of files, and 'TIDY' the files regularly. This will detect format errors, and serves usefully as another form of insurance against coding mistakes. It will also ensure that your files remain in operational order. Users (i.e. taxonomists or biologists, as opposed to computer experts) find it extraordinarily depressing to have a job abort because a computer claims to have detected more than 100 errors in their data. An unnecessary rise in blood pressure may result from failure to realize that gross insults of this kind may be responses to 'trivial' errors in your specifications, rather than to genuine defects in your data.

Use Intkey extensively — for example, to search for what you know to be unlikely or impossible combinations of characters. Obtain natural-language translations regularly, and make experimental conventional keys, in order to proof-read and search for nonsense. Testing keys is both entertaining, and effective in raising questions about the data and uncovering mistakes.

There is no need to become proficient in computer programming, nor even to be particularly knowledgeable about computers to become a successful user of the DELTA system; and computer-related chores should not usurp too much research time. Most taxonomists will need expert assistance in the initial stages of learning how to make files and use them, but once the system is running there should be relatively few demands on your consultant programmer. Most operations will require you to make quite simple modifications in the sample directives files which the system provides.

The [User's guide to the DELTA System](#) is very comprehensive, and if used properly will be found to carry the answers to most questions which are likely to arise. However, most potential users are unfamiliar with this type of literature. They will find it intimidating at first sight, and some perseverance will be required. Note that many concepts which are hard to explain in words are easily grasped via practical examples, with which the Guide is liberally illustrated.

## Appendix 2. Why must character states be mutually exclusive?

As mentioned in Appendix 1, the states of a character must be mutually exclusive: that is, no two of the states can both describe *the entirety of a particular instance* of the feature to which the character is applied. For example, the states of the character

```
#3. leaves <colour>/
  1. green/
  2. yellow/
```

are mutually exclusive because a leaf cannot simultaneously be yellow and green *all over*. Note that this doesn't preclude a leaf being partly green and partly yellow (3,1&2); or some specimens of a species having green leaves and others yellow leaves (3,1/2); or leaves being intermediate in colour (3,1-2).

On the other hand, the states of the ‘characters’

#2. leaves <yellow or narrow>/

1. yellow/
2. narrow/

#5. leaves <colour>/

1. green/
2. light green/

#6. leaves <length>/

1. 5-10 mm long/
2. 8-15 mm long/

are not mutually exclusive, because: a leaf can be both yellow and narrow; any leaf that is light green is also green; and a leaf that is 9 mm long could be described by either state 1 or state 2 of character 6.

To see the consequences for identification of states that are not mutually exclusive, consider the ‘character’ list

#1. leaves <colour, width>/

1. green/
2. yellow/
3. wide/
4. narrow/

#2. leaves <yellow or narrow>/

1. yellow/
2. narrow/

#3. leaves <colour>/

1. green/
2. yellow/

#4. leaves <width>/

1. wide/
2. narrow/

#3 and #4 are true characters, but #1 and #2 are not, because the states are not mutually exclusive.

Consider taxa A, B, C, and D, with the attributes

- A: leaves green, wide
- B: leaves green, narrow
- C: leaves yellow, wide
- D: leaves yellow, narrow

DELTA items describing these taxa in terms of the above ‘character’ list are

- # A/ 1,1/3 2<neither green nor wide recordable> 3,1 4,1
- # B/ 1,1/4 2,2<green not recordable> 3,1 4,2
- # C/ 1,2/3 2,1<wide not recordable> 3,2 4,1
- # D/ 1,2/4 2,1/2 3,2 4,2

where the comments have been added to indicate information that cannot be encoded in terms of that ‘character’.

Obviously, the four taxa are completely distinguishable using characters 3 and 4. But using ‘characters’ 1 and 2, the taxa are not completely distinguishable. For example, if we have a specimen of taxon A, we can only describe it (in an interactive or conventional key) in terms of character 1 (because neither state of character 2 is applicable. We might enter: ‘1,1’, which eliminates C and D, but fails to separate A and B; ‘1,3’, which eliminates B and D, but fails to separate A and C; or ‘1,1/3’, which eliminates D, but fails to separate A, B, and C. (Notice that entering *more* information (1,1/3) actually eliminates *fewer* taxa.) There are similar problems with specimens of the other taxa.

If the *author* cannot record (or chooses not to record) all the applicable states of 'characters' 1 and 2, use of these characters can lead to wrong results in identification, instead of just poor separability as above. For example, suppose that the author has only dry specimens, so that the colour is not known. The DELTA items would then become

```
# A/ 1,3 2,U 3,U 4,1
# B/ 1,4 2,2 3,U 4,2
# C/ 1,3 2,U 3,U 4,1
# D/ 1,4 2,2 3,U 4,2
```

If the *user* has a fresh specimen of taxon A, and enters '1,1' (leaves green), A is incorrectly eliminated.

Incorrect identification can also occur if the author adds a 'state' meaning 'unknown to the author' (instead of simply not recording the character, or recording it with the pseudo-value 'U'). For example, if leaf colour is unknown to the author, character 3 should be omitted or recorded as '3,U'. Instead, authors sometimes define a 'character' such as

```
#7. leaves <colour>/
  1. green/
  2. yellow/
  3. unknown/
```

and record it as '7,3' if the colour is unknown *to them*. However, the user has no way of knowing that this has been done, and may enter '7,1' or '7,2', which will cause the taxon to be incorrectly eliminated.

## Acknowledgements

We thank Marianne Horak and Max Day for reviewing this document, and suggesting many improvements.

## References

- Dallwitz, M.J. 1974. A flexible computer program for generating identification keys. *Syst. Zool.* 23: 50–57.
- Dallwitz, M.J. 1980. A general system for coding taxonomic descriptions. *Taxon* 29: 41–46. Also available at [delta-intkey.com](http://delta-intkey.com).
- Dallwitz, M.J. 1992 onwards. Overview of the DELTA System. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J. 1997 onwards. DELTA programs and documentation. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J. 2016 onwards. Programs available in the DELTA System. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J., Paine, T.A., and Zurcher, E.J. 1993 onwards. User's guide to the DELTA System: a general system for processing taxonomic descriptions. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J., Paine, T.A., and Zurcher, E.J. 1995 onwards. User's guide to Intkey: a program for interactive identification and information retrieval. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J., Paine, T.A., and Zurcher, E.J. 1999 onwards. User's guide to the DELTA Editor. [delta-intkey.com](http://delta-intkey.com)
- Dallwitz, M.J., Paine, T.A. and Zurcher, E.J. 2000 onwards. Principles of interactive keys. [delta-intkey.com](http://delta-intkey.com)
- Partridge, T.R., Dallwitz, M.J. and Watson, L. 1993 onwards. A primer for the DELTA System. [delta-intkey.com](http://delta-intkey.com)

## Citation

Publications making use of the components of the DELTA system must cite as follows.

Components used	Citation
DELTA Editor	Dallwitz 1980 Dallwitz, Paine, and Zurcher 1999 onwards
DELTA format, Confor, Dist	Dallwitz 1980 Dallwitz, Paine, and Zurcher 1993 onwards
Key	Dallwitz 1974 Dallwitz 1980 Dallwitz, Paine, and Zurcher 1993 onwards
Intkey	Dallwitz 1980 Dallwitz, Paine, and Zurcher 1993 onwards Dallwitz, Paine, and Zurcher 1995 onwards Dallwitz, Paine, and Zurcher 2000 onwards

Versions of documents published on [delta-intkey.com](http://delta-intkey.com) are identified only by the date of publication. If you want to cite a particular version, use ‘... Version: *date*. [delta-intkey.com](http://delta-intkey.com)’, where ‘*date*’ is the date at the top or bottom of the document.

### *Example*

Dallwitz, M.J., Paine, T.A., and Zurcher, E.J. 1999 onwards. User’s guide to the DELTA Editor.  
Version: 10th September 2016. [delta-intkey.com](http://delta-intkey.com)

Please send the reference for your publication to [M.J. Dallwitz](mailto:M.J.Dallwitz), who will add it to the [list of DELTA publications](#) on the Web.