ZooKeys 45: 1–75 (2010)
doi: 10.3897/zookeys.45.263
www.pensoftonline.net/zookeys

REVIEW ARTICLE

A peer-reviewed open-access journal
ZooKeys
Launched to accelerate biodiversity research

DELTA for Beginners: An introduction into the taxonomy software package DELTA

Ch. Oliver Coleman[1], James K. Lowry[2], Terry Macfarlane[3]

1 *Museum für Naturkunde, Berlin* **2** *Australian Museum* **3** *Western Australian Herbarium*

Corresponding author: *Charles Oliver Coleman* (oliver.coleman@mfn-berlin.de)

Academic editor: *Lyubomir Penev* | Received 31 August 2009 | Accepted 30 November 2009 | Published 05 May 2010

Citation: Coleman CO, Lowry JK, Macfarlane T (2010) DELTA for Beginners: An introduction into the taxonomy software package DELTA. ZooKeys 45: 1–75. doi: 10.3897/zookeys.45.263

Abstract

An introduction to the software package DELTA (**DE**scription **L**anguage for **TA**xonomy) is given. The contribution consists of step by step instructions into the DELTA Editor and the interactive identification program Intkey. It describes how to record taxonomic character information in a database and maintain these data. Standard output functions are simplified in a new starter database. All used commands are commented and it is marked where changes in the command files are required. The paper explains how to generate text descriptions, interactive identification tools, and how to make keys and species diagnoses.

Keywords

DELTA, software, taxonomy, database, systematics, descriptive language for taxonomy, biodiversity informatics

Introduction

Situation of systematics

Taxonomy in the early 21st century is in decline. Morphological systematics is no longer part of the university curriculum. Its role in large government institutions such as the Australian CSIRO is minimal and even in its most secure environment, a natural history museum, it is threatened. In 1990 the Natural History Museum of the Smithsonian

Institution had one of the largest groups of invertebrate taxonomists the world has ever seen, 25 scientists (including those from the Smithsonian Oceanographic Sorting Center). In the year 2007 there were only 15 left. In the Crustacea department of the Smithsonian the number of curators shrunk from 8 to 3 in the same time interval. On a smaller scale, during the 90's, the Australian Museum had a usual staff of about 24 scientists. In 2007 the number of scientists has been reduced to 12.

During a time when cataloguing and studying the world biota is crucial, the number of world taxonomists has been halved.

Funding agencies appear to view taxonomy as an unsophisticated science with outdated methodologies. But the fact is that if we don't have well catalogued, well described biota we cannot make rational decisions about our environment. Taxonomy is a science we cannot do without.

Efficiency and consistency is particularly important in taxonomy, especially when taxonomists may be describing species in the same genus from opposite sides of the world. The software package DELTA is a sophisticated and powerful databasing program which stores morphological data for export in a number of different forms. In this way it acts as a manager of taxonomic research which can be used on a local, regional or world wide basis to make biota available to whomever needs to use them.

Why use DELTA?

DELTA is a modern software package that manages taxonomic research (Dallwitz 2009):

- It promotes consistency in descriptive taxonomy.
- It defines taxonomic terminology.
- It can be used alone or shared among groups of taxonomists.
- It can be passed on to your taxonomic progeny when you die.
- It is long term. Databases can start small and develop to cover world faunas.
- It can produce descriptive taxonomy in natural language ready for publication.
- It can produce conventional ('dichotomous') keys.
- It can produce interactive identification systems for private use or use on the web or CD.
- It can produce data matrices for phylogenetic analyses and manage the data for phylogenetic projects.
- It can be used to build small databases to the taxa of an area or the species in a genus or larger databases to species of higher level taxa.

Using DELTA allows a taxonomist to build and store the core of his/her research over many years. It can be continuously modified and updated to capture local, regional or world faunas. Databases started by taxonomists in one region can be passed to taxonomists in other areas to produce morphological databases for world faunas.

New taxa can sit beside known taxa within databases to be used, and then published when appropriate.

At times the information in these databases can be transformed and exported into Intkey, which produces interactive identification systems for private use or web use.

Finding new species

DELTA databases are important tools for analyzing large numbers of taxa. By using the tools in Intkey a taxonomist can carefully find and evaluate morphological differences. In this way taxa hidden in the literature or from collections can be identified, described and published.

When starting to build a character list it is often useful to use the characters in dichotomous keys to the taxa you are interested in. You might as well utilize the already accumulated information. However, the data in dichotomous keys is sparse, and must be supplemented in order to make satisfactory descriptions, classifications, and interactive keys.

How to start

Obtaining DELTA

The software package DELTA can be downloaded for free from the website http://delta-intkey.com (search under: *Programs and documentation, Programs, All programs (including Intkey)*); clicking on the last mentioned link will produce a download dialog and as selected by the user, the installation program *delt32.exe* (5.7 MB) will be downloaded or run. The programs, documentation, and sample data will then be installed onto the harddisk, usually in the folder c:\delta.

Running DELTA under MS Vista

Installing DELTA under Vista can cause difficulties. If this happens, install the full DELTA package, there will be an error message, break off the installation.

1) install Intkey, there should not be a problem to install it.
2) DELTA should be installed on another computer running Windows XP. Then copy the DELTA directory (c:\delta) onto a external drive (e.g. USB-drive) or CD and copy the contents of the DELTA folder from this onto the machine running Vista in c:\delta. The DELTA Editor (delta.exe) will not initially start without prompting. Right-click on delta.exe in the (DELTA directory), then select “Run as administrator”

and acknowledge that the program can be trusted. Hereafter the program will run without difficulties. Intkey will start normally without such secondary prompting.

User guides

In addition to this introduction to DELTA, information about preparing DELTA databases can be found in the User's Guide to the DELTA Editor (Dallwitz et al. 1999) and the User's Guide to the DELTA System (Dallwitz et al. 1993). Note that the versions included in the program installation are obsolete, and should not be used.

The User's Guide to the DELTA Editor describes a program for entering and maintaining DELTA data.

The User's Guide to the DELTA System is a reference manual for:

- The DELTA data format (Dallwitz 1980).
- The program Confor, for converting DELTA-format data into natural-language descriptions, and into the formats required by various other programs.
- The program Key, for generating conventional ('dichotomous') keys.
- The program Dist, for generating distance matrices.
- How to set up Intkey datasets on the web.

An Intkey Example: Identification (Dallwitz 2000) is an introduction to using the interactive-key program, Intkey.

We recommend that new users work through these examples. However, we do not recommend that users construct their own database from the sample data.

Creating a work directory

Before starting to work, the user has to copy a database (e.g. *Deltablank.dlt*) into a separate directory or subdirectory. In the course of the work this directory will be filled with many files generated by export procedures from DELTA. Using the original DELTA folder for this could result in the loss of some important files.

Sample data file vs. *Deltablank.dlt* template

A sample database on grass genera is included in the DELTA installation. It is found in the subfolder *sample* in the DELTA folder. This can be used to take the first steps in DELTA and try out the great possibilities of the software.

Deltablank.dlt is based on the 'New Dataset' of the DELTA Editor, but makes the creation and use of a new database easier for beginners in the following ways.

(1) The following "action sets" (see later), which we consider less likely to be required by beginners, are omitted. (a) Format the character list as HTML (for publication on the Web). (b) Translate into natural language in HTML format. (c) Produce conventional ('dichotomous') keys. (d) Translate into Nexus format (for cladistic analysis). (e) Produce a distance matrix (for phenetic analysis). These "action sets" can be copied from a standard, empty DELTA database if required later.
(2) In the action sets, extensive comments have been added to more fully explain the directives.
(3) This paper describes the processes of creating and using a database in more detail than the User's Guide to the DELTA Editor.

The *Deltablank.dlt* template can be downloaded here: http://download.naturkunde-museum-berlin.de/oliver.coleman/delta/.

For experienced users: At a later stage the user can start a new database in the DELTA Editor by selecting *File / New Dataset*. The database so created is empty except for several 'actions sets', which contain 'directives' (i.e. commands) specifying ways in which the data (when it is eventually entered) can be processed. The empty database is based on the DELTA sample data (not on the Deltablank template mentioned above and used throughout this paper), but omits the character list and taxon descriptions. In the action sets, directives (commands) that might be incompatible with the new data are deactivated, but they can be modified to suit the new data, and then activated (see below). The section 'Creating a Data Set' in the User's Guide to the DELTA Editor contains basic instructions for entering data into the empty database.

Restrictions on file names

Some components of the DELTA software package date back to the time of the MS-DOS operating system. Thus, like in the early days of personal computing, file names for the output files and subdirectories are restricted to 8 characters (except for image files which can be longer) plus a 3 character extension (file type). Also, file and folder names used by the programs must not contain blanks or other special characters such as "**.**".

Terminology

DELTA has its own terminology: items = taxa; features = characters; directives = commands; Action sets = output presets.

The term *item* is used because a taxon may occur in more than one form. For instance male, female, or growth stage. DELTA can handle any of these forms and each one becomes an item under the same taxon.

The DELTA Editor

The Editor is the main program of the DELTA package. Many of the other programs in the DELTA package can be accessed by the Editor.

The Interface

As with most Windows programs, it is possible to open a file in the DELTA editor in two ways:

1. By running the DELTA Editor from the Start menu, and then opening a database by selecting 'Files / Open Dataset'.
2. By double clicking on a DELTA database in Windows Explorer. The databases are recognisable by the extension .dlt (unless extensions have been hidden in the Folder Options), or by a DELTA icon '  '.

If the database is new, as it would be when loading the template *Deltablank.dlt*, the user would see four empty panes (Fig. 1). The upper left pane will contain the taxa (or items), the upper right pane is for the characters. In the lower right pane the character states will appear and in the lower left pane the selection of character states in numbers. It is possible to adjust the size of these panes by clicking with the left mouse button on the separation bars, keeping the mouse button pressed, and then drag and drop them.

The dropdown menu entry *File* allows the user to manage databases (access and saving of databases, importing and exporting of directives [about export of directives see below]) (Fig. 2a). Under *Edit, Search* and *Window* (Fig. 2b, c, e) the user will find standard functions known from other Windows programs. Under *View* (Fig. 2d) some very important DELTA functions are listed: the *Attribute editor* in *Tree* form is the view of the DELTA Editor shown in Fig. 1. The *Attribute editor* in *Grid* view shows taxa, characters and character states in a spreadsheet format. *Character editor* and *Taxon editor* will be explained below. An important and frequently accessed menu selection is *Action sets* which includes all output options (e.g. output of natural language descriptions and creation of interactive keys). A whole section in this paper is devoted to the *Action sets*. A following chapter is devoted to *Action sets*. The *Image settings* control the handling of images and text overlays that will appear in *Intkey.* The *Help* dropdown

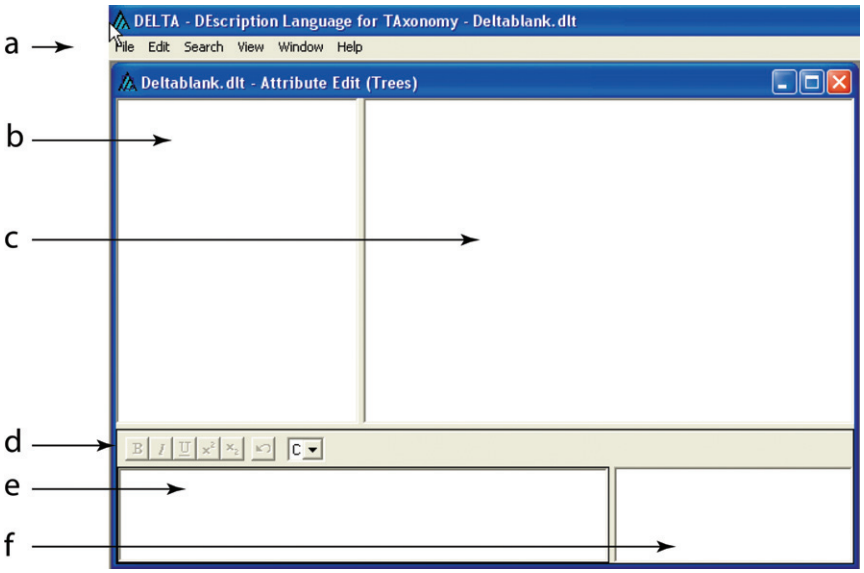


Figure 1a–f. DELTA Editor. **a** dropdown menus **b** pane for items (= taxa) **c** pane for characters **d** buttons for formatting selections **e** pane showing the character state description in number form and possible additional descriptions in angle brackets **f** pane for character state selection with check boxes.

menu provides an *About* option; the user should refer to the User's Guide to the DELTA Editor for help. A row of small buttons (Fig. 2g) assists the user to format the text.

The Taxon editor allows the input of data related to each individual taxon (or item). It can be called up from the dropdown menu (*View/Taxon editor*) (Fig. 2d) or more conveniently by double-clicking on the items (taxa) pane, the upper left pane in the standard view of the DELTA Editor (Fig. 1b). A *taxon number* (Fig. 3a) is automatically assigned. The arrow buttons can be used to browse the taxa. The field *Edit taxon name* (Fig. 3b) must contain a taxon name. Comments can be made in angle brackets. This name will appear in the items (taxa) list (Fig. 1b).

There is a checkbox, *Treat as variant* (Fig. 3c), that allows the user to enter only those attributes that differ from the preceding 'main' item. These might be differences in growth forms, populations or sexual dimorphism. Except in natural-language descriptions, the programs automatically fill in the unrecorded attributes in the variant item from the corresponding ones in the main item. When the variant item box is checked, a + sign appears before it in the list of items. It is possible to link illustrations and sounds to the taxa (Fig. 3d, e). These are used in Intkey and in natural-language descriptions. A row of formatting buttons for the text is provided (Fig. 3f).

The Character editor looks very similar to the Taxon editor. It can be called up from the dropdown menu (*View/Character editor*) or by double-clicking on the character list, the upper right pane in the standard view of the DELTA Editor (Fig. 1c). A number is automatically assigned for each character (Fig. 4a) and the arrow and *Select* buttons can be used for browsing through the characters. The 'feature' to which the

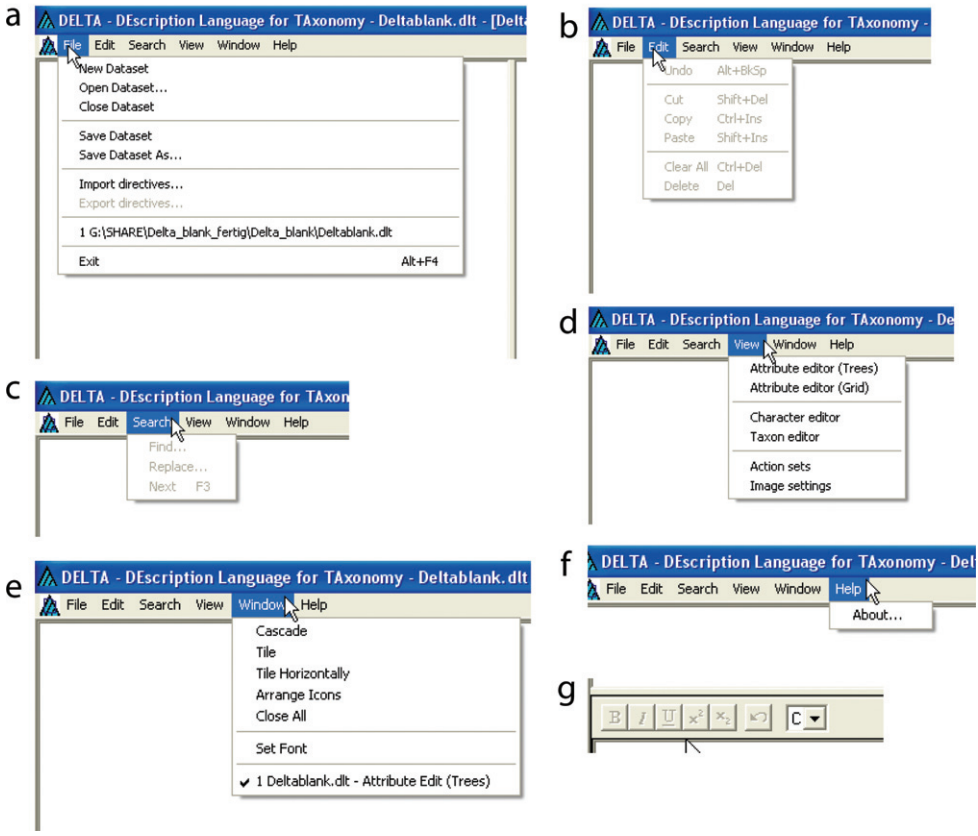


Figure 2a–g. Dropdown menus of the DELTA Editor.

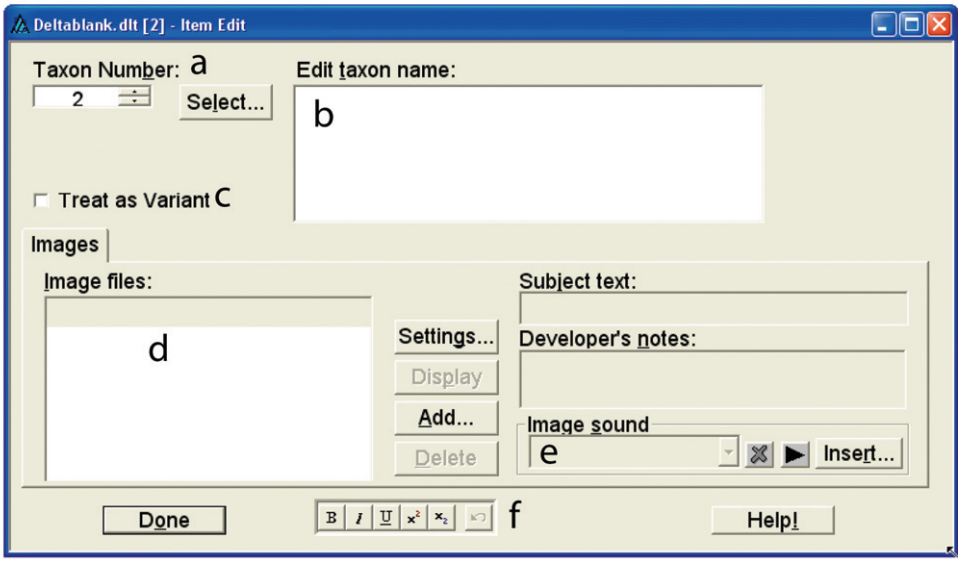


Figure 3a–f. Taxon editor, explanations in the text.

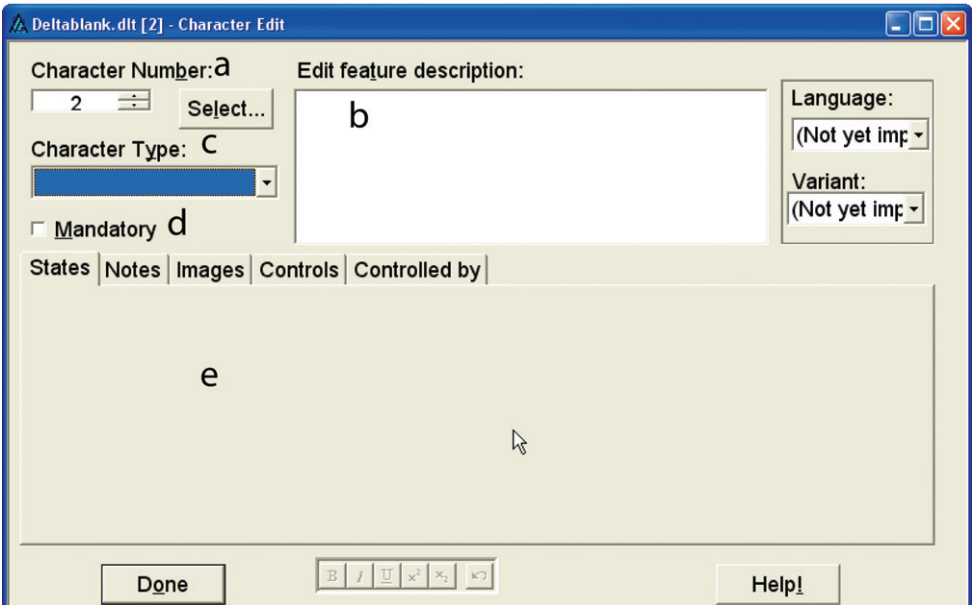


Figure 4a–e. Character editor, explanations in the text.

character applies (e.g. head, eyes, mandibles) is entered in the Edit feature description box. The character state and other useful options are available behind the tabs in the lower half of the Character editor (Fig. 4e). How character states are defined will be described in the next subsection. If a character is important it can be designated *mandatory* (Fig. 4d). An error message would come up if a mandatory character is not coded in any of the taxa (items).

Input of Data

Input of characters and their states

It is wise to compile a list of characters before actually starting to code the character states for the taxa. Published dichotomous keys and descriptions provide a good set of useful characters that can be used in a new DELTA database.

Character types

Several types of character are supported by DELTA. These can be selected in the Character editor under *Character Type* (Fig. 4c). If the user clicks on the arrow button, a dropdown menu will open and the selection can be made (Fig. 5).

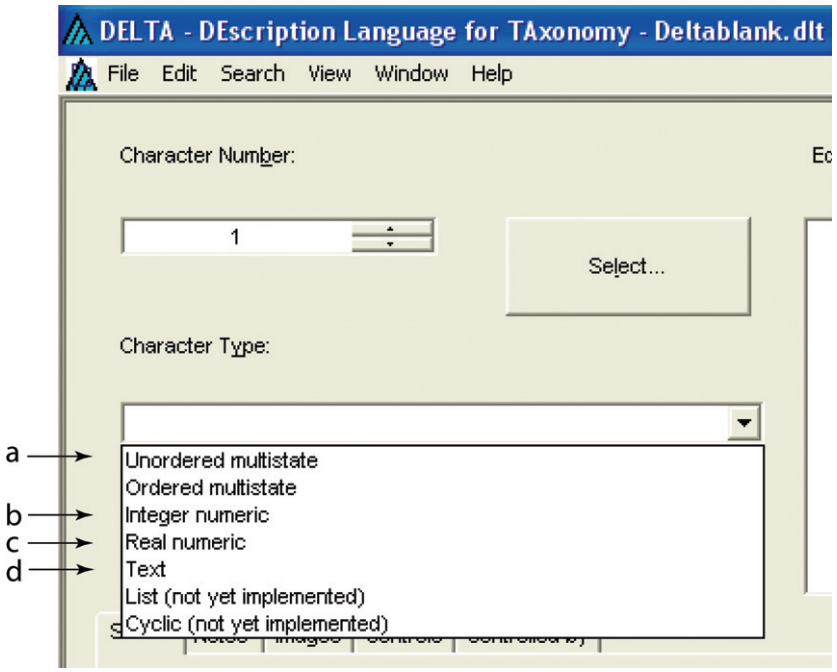


Figure 5a–d. Character types selection from the Character editor.

The *Unordered multistate* character type (Fig. 5a) is used for characters where the states are not arranged in a natural order. *Ordered multistate characters* are used for characters whose states have a natural order. They may contain hypotheses about the evolution of a character and are mainly used for creating phylogenetic matrices. It is not treated herein. There are two types for the input of numbers (Fig. 5b, c). *Integer numerics* are used for counts (e.g. 4, 6, 11). *Real numerics* can be used for measurements. They can contain fractions (e.g. 1.2; 5.65). After selecting the *Text* type (Fig. 5d) the user can store pure text information for each taxon, for example synonymy, common names, manuscript names, etymology, station data, distribution data and discussion.

Text characters

We now enter some text characters before we come to multistate characters and numerics. We insert 'taxon name' in the *Edit feature description* box (Fig. 4b) and include this feature in angle brackets (Fig. 6a). This prevents the output of this text element prior to the actual taxon names in the generation of natural language descriptions or interactive keys. However, in the next text character 'fig.' this part should appear in the description and thus there are no angle brackets (Fig. 6b). In Fig. 6c there is already a longer list of text characters, but character 7 should rather appear right after character 4. So we click with the left mouse button on character 7, hold the button and drag the

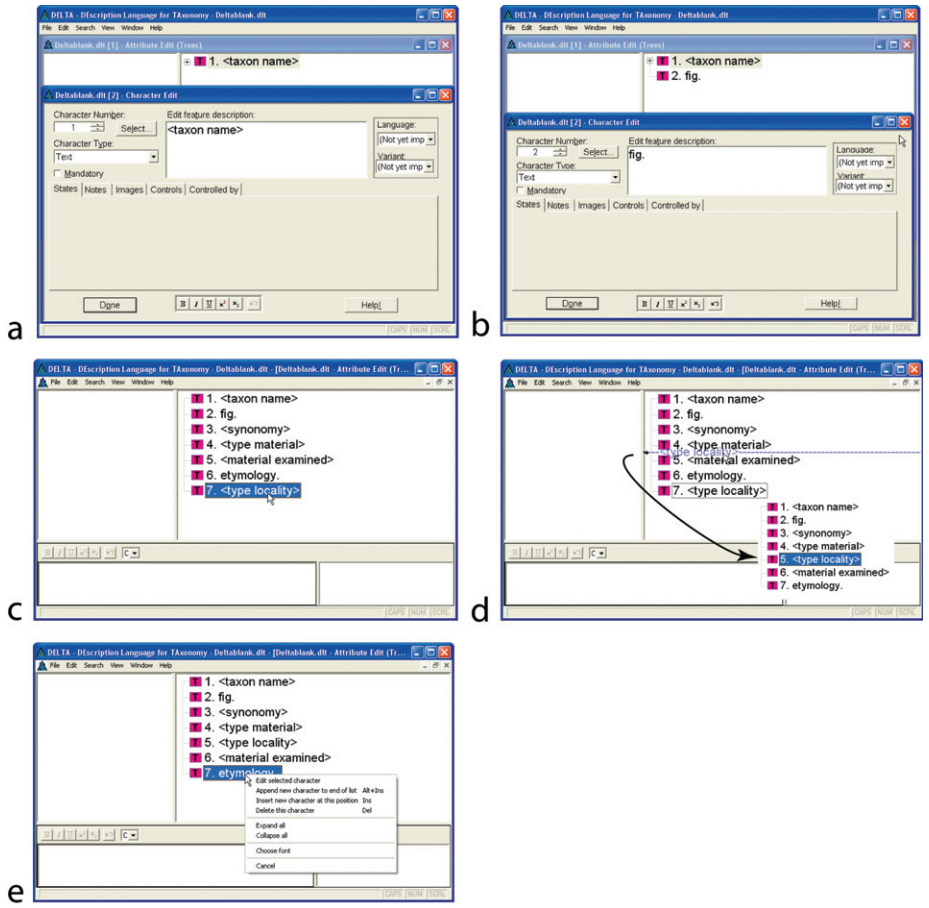


Figure 6a–e. Input of text characters with the Character editor and change of sequence of characters.

text line right after character 4 (Fig. 6d) and drop it there (arrow points to result). The character numbers are updated automatically. If we right-click on one of the characters (as in Fig. 6e) the dialog enables us to insert a character just prior to (in our example selected) character 6.

In our example there are several feature descriptions for characters in angle brackets, - for good reason. We will show later how feature descriptions for characters 4-6 are replaced by bold headings in the natural language outputs.

Multistate characters

Adding *Unordered multistate characters* (see Fig. 7a–b)*:*

1) By double-clicking on the character window (the upper right pane) of the DELTA Editor the *Character editor* window will open.
2) From *Character Type* the *Unordered multistate* type is selected.

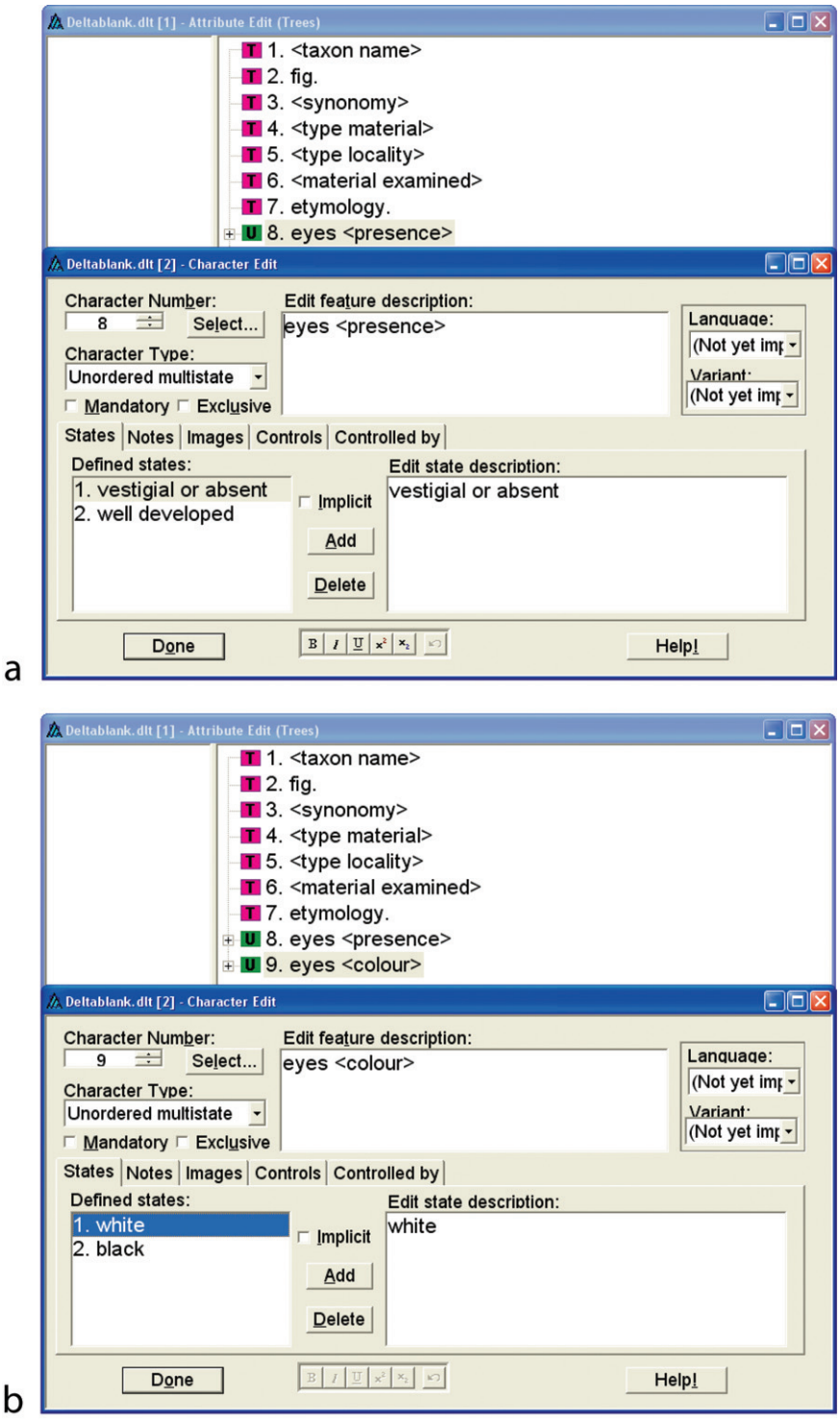


Figure 7a–b. Definition of two *Unordered multistate characters*.

3) In *Edit feature description* the feature text '*eyes <presence>*' is inserted (Fig. 7a). The 'comment' text in angle brackets should make the feature description a brief summary of the character, so that the character can be recognized from the feature description alone. It is important for recognizing the character in some contexts, such as the characters pane and the 'grid' Attribute Editor in the DELTA Editor, and in Intkey. The comment text is omitted from natural-language descriptions.
4) In *Edit state description* the text of the first character state '*well developed*' is put in, then after pushing the *Enter* key, the next character state '*vestigial or absent*' is inserted.
5) The sequence of the character states in *Defined states* can be changed by using the drag-and-drop function with the mouse.
6) If '*well developed eyes*' in our example are the rule and the '*vestigial or absent*' state is the exception, then we could select the *implicit box* for the first mentioned character state. When recording the taxa the implicit state will be shown as a default selection, and the state will be omitted from natural-language descriptions (unless recorded explicitly). We shall explain later how state values are recorded for taxa.
7) When all character states are inserted, the *Done* button will close the *Character editor*. Or, if characters are being added to the end of the list, one can enter the next one without closing the *Character Edit* dialog, after selecting the next *Character number*.
8) We put in another feature '*eyes <colour>*' (Fig. 7b) with the character states '*black*' and '*white*' in order to demonstrate another important aspect, the C*ontrolling attributes*, which are explained in the next chapter.

Character notes

Under the tab *Character notes* within the *Character editor* the user can explain details of character which will be displayed in Intkey.

Controlling attributes

In our example it does not make any sense to ask for the colour of the eyes when the eyes are vestigial in a taxon. In DELTA there is a function for dealing with such cases (Fig. 8a):

1) In our example we open the *Character editor* by double-clicking on character 8 ('*eyes <presence>* '),
2) select the tab *Controls,*
3) select '*vestigial or absent*' and click on *Define*, the selected character state appears in *Controlling attribute* field*,*
4) click on the character in the right field (*Character list*) that should be made inapplicable and

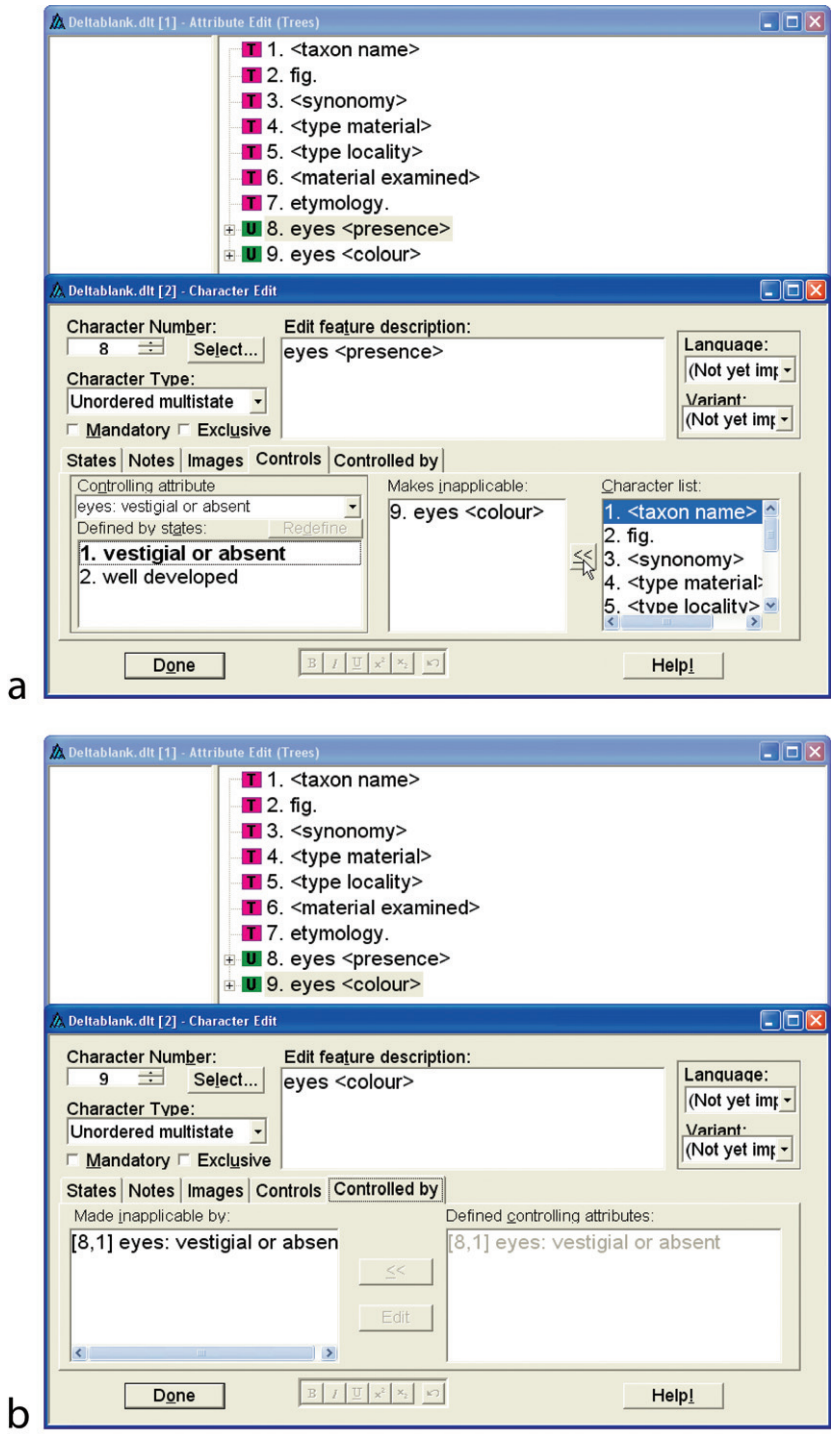


Figure 8a–b. Controlling character states.

5) transfer it to the field in the center of the dialog (*Makes inapplicable*) by selecting the button with the *double-left-arrowheads*.
6) In character 9 we can check the dependencies opening the *Controlled by* tab (Fig. 8b). The *Made inapplicable by* field shows that [8,1] (= character 8 – character state 1), '*eyes: vestigial or absent*' controls the selected character '*eyes <colour>'*.
7) When recording taxon descriptions, in the character list an inapplicable character is marked with a red slash on the coloured character type symbol.

Numeric characters

There are two kinds of numerical characters, integers and real numbers.

The *Integer numeric* is a whole number. In our example (Fig. 9a) the DELTA Editor asks for the number of articles in antenna 1

1) select *Integer numeric* as the *Character type* and
2) write the feature text '*antenna 1 with <number of articles>*' in the field *Edit character description* of the Character editor, then
3) insert '*articles*' in *Units.*
4) So the output for a species with 5 articles would read: '*Antenna 1 with 5 articles*'. (We will come back to output when we discuss the generation of natural language descriptions).

The *Real numeric* character type allows the input of fractions of a number. Character 11 in our example is such a case (Fig. 9b). We

1) select *Real numeric* as *Character Type* in the Character editor and
2) insert the character text '*antenna 1 article 1 <length – width ratio'* in the *Edit feature description* box,
3) in *Units* we put in '*x as long as wide*'.
4) After measuring our taxon and recording the measurement the output would read: '*Antenna 1 article 1 3.2 x as long as wide*'.

Taxa

Inserting additional taxa into the database is simple. Double-clicking in the empty part of the taxon pane (in the upper left of the DELTA Editor) opens the *Taxon editor* (*Item Edit*) (Fig. 10) and a taxon name can be put in the *Edit taxon name* box. After clicking on the *Done* button the name of the taxon is displayed in the taxon pane. If you have already put in several taxa, right clicking on a taxon allows a new taxon to be inserted below the selected taxon or at the end of the list. Double-clicking on a taxon opens the Taxon editor for making changes or additions.

You can record the character states in the check boxes in the lower right pane of the DELTA Editor (Fig. 11).

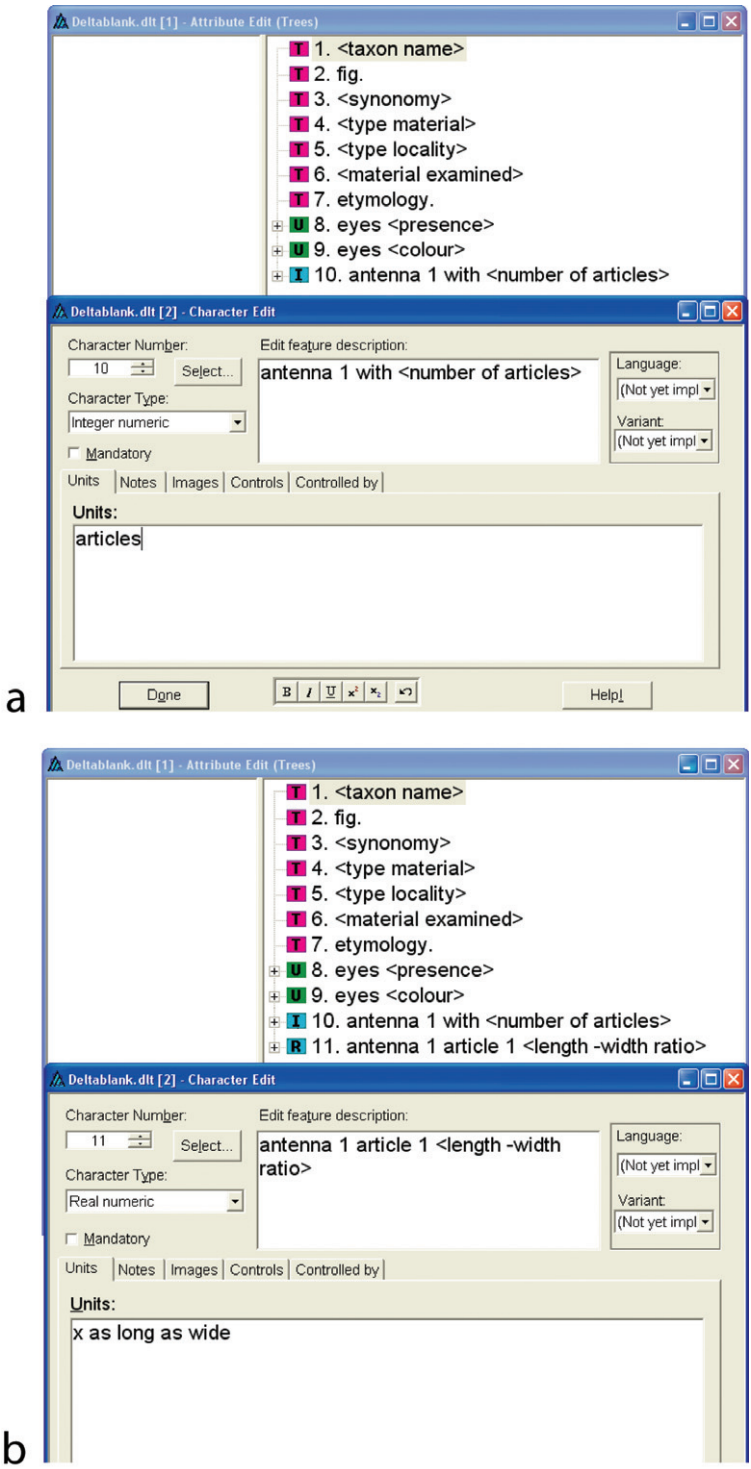


Figure 9a–b. Numeric character types. Examples of **a** *Integer numeric* **b** *Real numeric*.

In Fig. 12 we opened the character list in the tree-like view by right-clicking in the pane and selecting *Expand all* and coded the character states directly in this pane.

A white background in a character-value check box, indicates that the attribute is 'simple', and the value(s) can be edited in situ. A grey background 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.

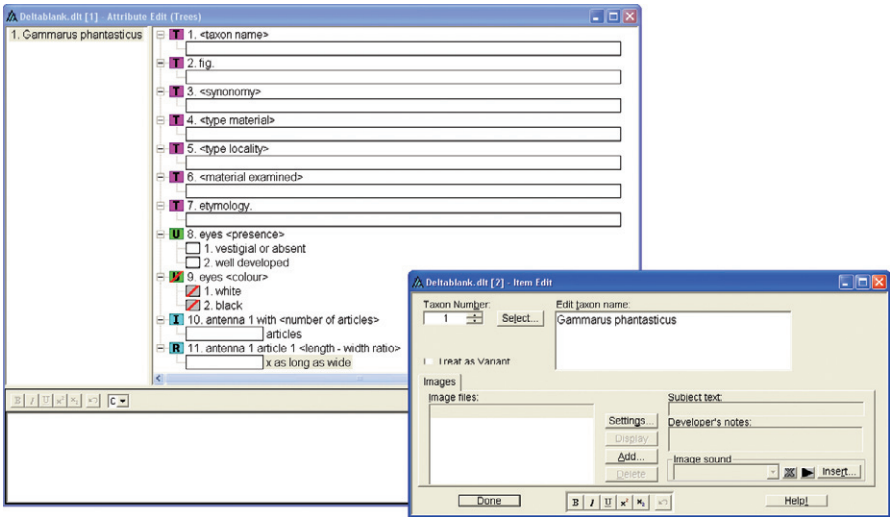


Figure 10. Inserting taxa in the DELTA Editor.

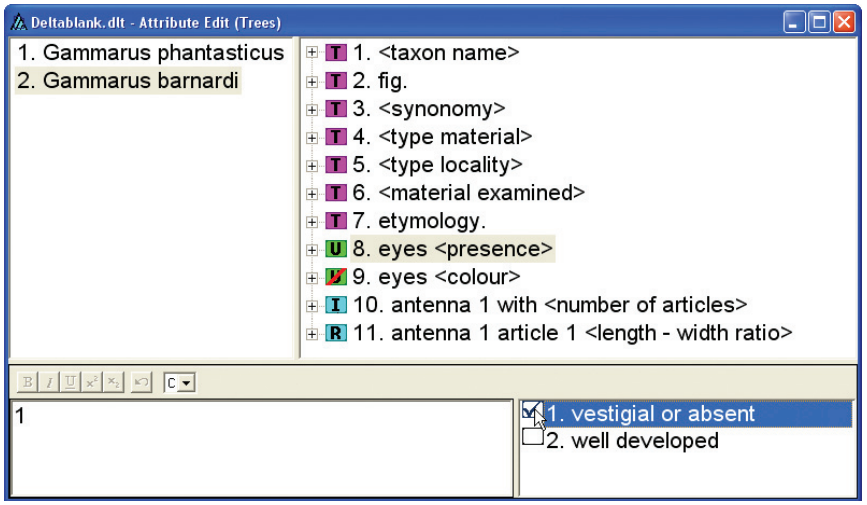


Figure 11. Recording the characters using the check boxes for states in the lower right pane.

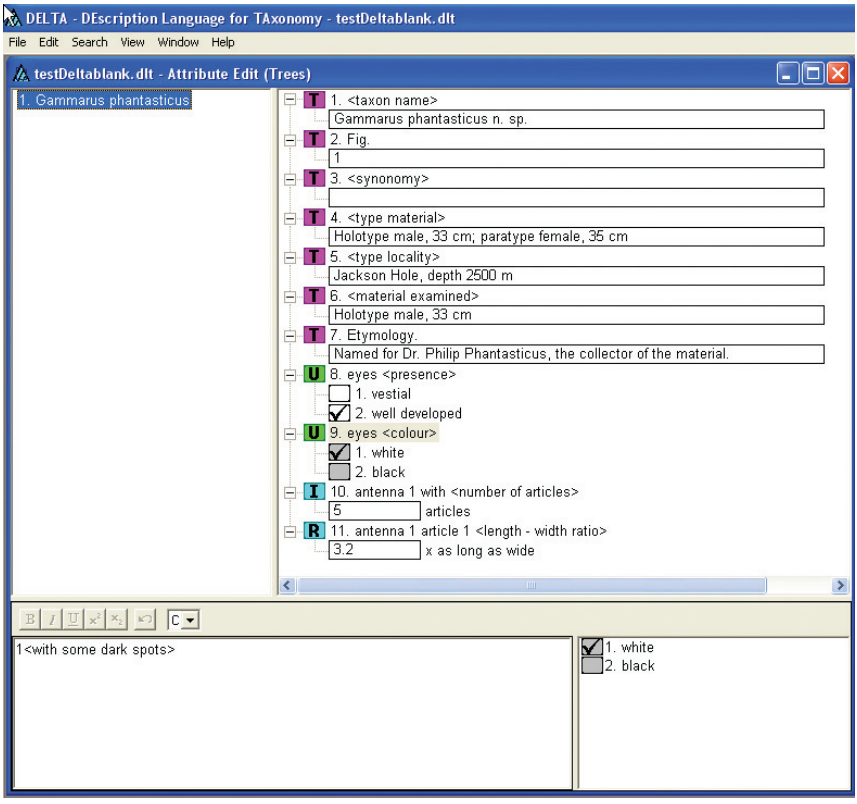


Figure 12. Recording the character states in the tree–like expanded character pane. It is possible to add comments in angle brackets, and other qualifying information, in the lower left pane.

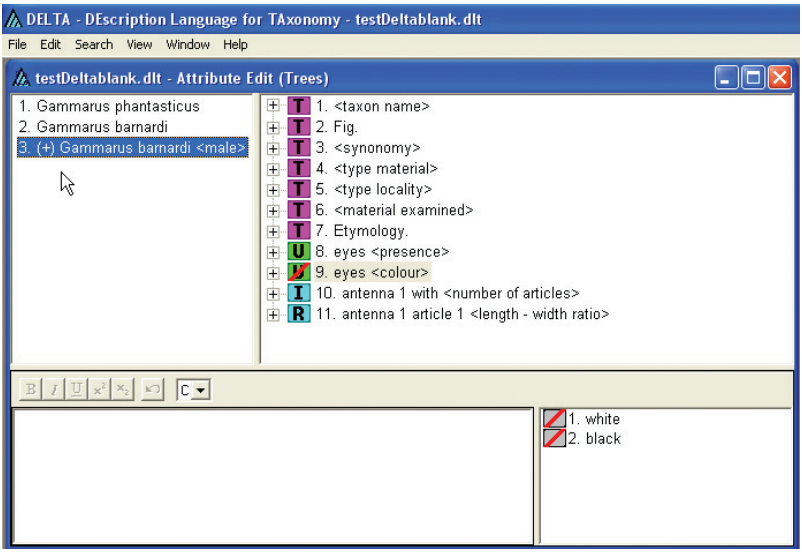


Figure 13. Inserting a taxon that was marked as a variant (*Treat as variant*) in the Taxon editor, indicated by (+) preceding the taxon name.

Complex attributes can be edited or entered only in the complex-attribute editing pane at the bottom-left of the screen. Comments qualifying a state are entered in angle brackets immediately after the state number (Fig. 12).

A taxon *Gammarus barnardi* <male> was added as a variant (Fig. 13). This was accomplished by checking the box *Treat as variant* in the *Taxon editor*. The (+) symbol indicates the variant condition. Using variants allows the output of characters that differ from the original taxon in natural language descriptions and avoids the repetition of redundant information.

Moving of taxa within the taxon window is possible by dropping and dragging as described above for characters.

Attribute editor in Grid view

Besides the default *Tree view* shown above, it is possible to view and edit the DELTA database in a grid view (Fig. 14), selected from the dropdown menu *View/Attribute editor (Grid)*. Like in a spreadsheet, taxa (Fig. 14b) and characters (Fig. 14a) are in vertical columns and horizontal rows and character states are shown as numbers in the grid. Unknown characters can be explicitly marked as unknown with an '*U*' (works

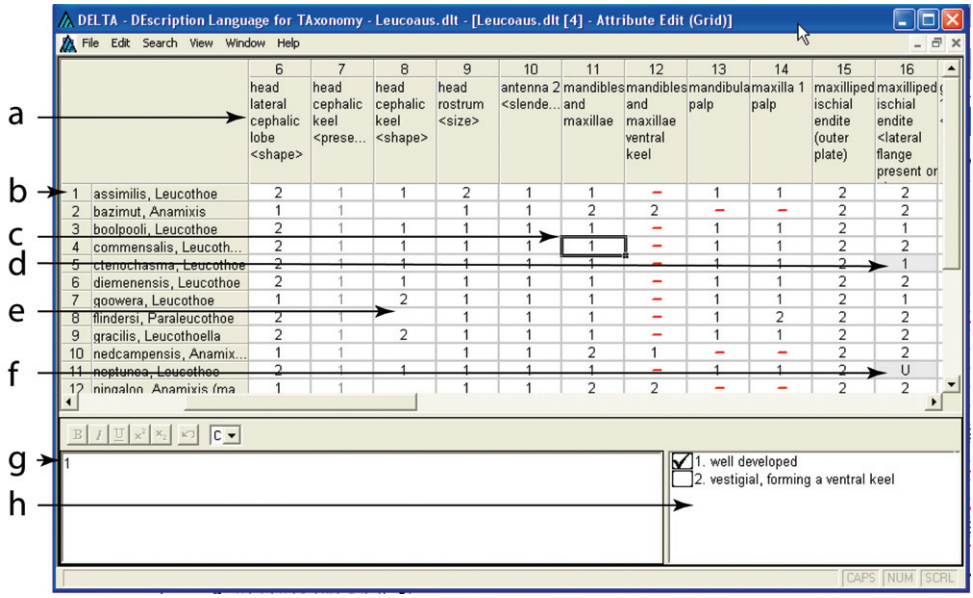


Figure 14a–h. Attribute editor in the Grid view. **a** character feature text **b** taxon list **c** selection box shows character state 1 selected. The red dash of character 12 indicates that this character is inapplicable **d** the grey background in the box indicates that a comment in angle brackets was made to specify the character state **e** unrecorded character **f** character states explicitly marked unknown **g** character state pane, where aside from the input of the character state as a number comments can be made in angle brackets **h** character state selection check boxes.

also in the *Tree view*) (Fig. 14f) or simply not scored (Fig. 11e). In character 7 in our example in Fig. 14, the character state '*1*' is in grey. This means that this character state is marked as implicit (see *Implicit* checkbox in the *Character editor*), the default state for the character.

Wording of characters

1) It is wise to avoid too many character states for a character. It is much better to split up complex characters into several simple ones.

2) In characters which form a complex, the feature texts should start with the same words. If these characters are marked as groups in the *layout* Action set (we come to this later) then the output will avoid the repetition of the first text element.

For example characters 6-9 in Fig. 15 start with '*head*'. This part of the text occurs only for the first character and are omitted in the following ones. So '*head*' would occur only once for the whole block of characters. After grouping characters 6-9 in the *layout* Action set (see below) the text output would read as follows: '*Head lateral cephalic lobe without cusp; cephalic keel present, cephalic keel similar in size to epistome; rostrum small to medium in size.*' As default the semicolon is used to separate the characters in a group. Semicolons can be changed in the *layout* Action set into commas. In our example one of the semicolons was replaced by a comma, connecting the two '*cephalic keel*' characters.

3) It is important to add comments (in angle brackets) to the feature descriptions so that the nature of a character is summarized by the feature text. If for example the number of articles for a palp is recorded, then the feature text should be for example: '*palp <number of articles>*'. Otherwise the user would have to look at the

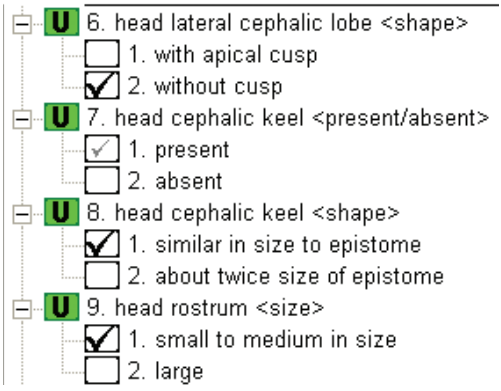


Figure 15. Example of characters forming a group where certain text elements are repeated.

character states in order to find out what the feature character describes. These comments are especially important for the use in Intkey. Also the taxon text, editable with the Taxon editor (inserted in the *Edit taxon name* box), can include comments, e.g. '*Leucothoe spinicarpa <deep water population>*'.

In the first case the comment between angle brackets is suppressed in the generation of natural language, in the case of the taxon it remains. Also angle bracket comments can specify character states of individual taxa (to be put in the lower left pane of the DELTA Editor following the character state number without a space) and can be put out in text descriptions.

4) Comments in double angle brackets may be (and normally are) omitted from natural-language descriptions.

Using illustrations

Taxon images

Including illustration for taxa is very easy. In *Settings* within the Taxon editor (Fig. 16) we specify the path where the images are stored. By *Add* or *Delete* the selection of images from the folder is carried out. The selected images are listed in the dialog box *Image files* (Fig. 16). After double-clicking the file names the illustrations will be added to the list. These image files are also available in Intkey.

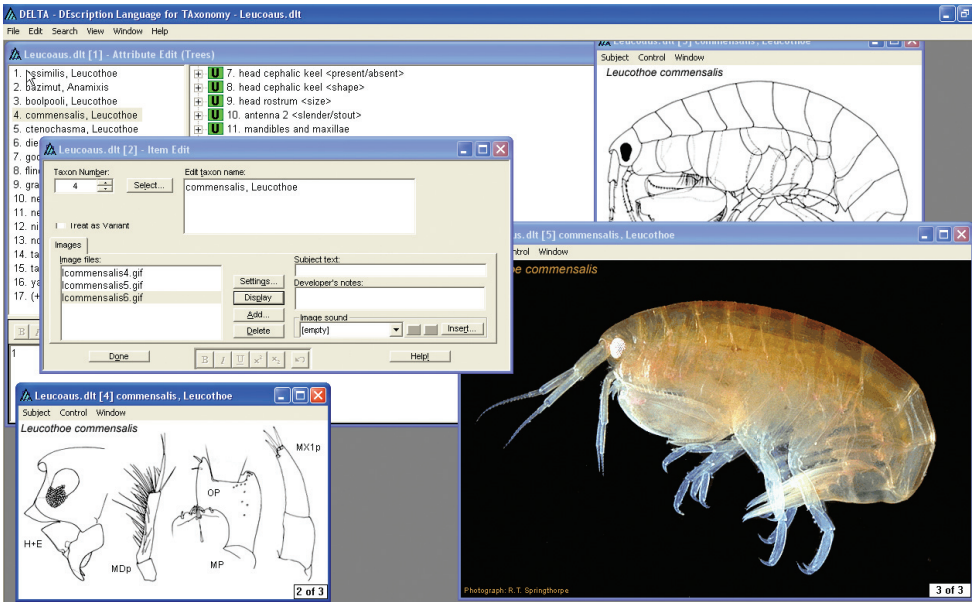


Figure 16. Selection of taxon illustrations from the Taxon editor.

Character images

Analogous to the Taxon images the Character state images are accessible through the Character editor (*Images* tab in Fig. 17a). In *Settings* (Fig. 17b) the path to the directory where the images are stored can be specified and font details for the overlays (see below) can be selected. By *Add* and *Delete* the images for the character are selected from the image folder and are listed in the box *Image files*. An image can be displayed via the Display button, or by simply double-clicking on the file name.

Creating overlays and hotspots

In the displayed illustration appear several boxes (so-called overlays) which are used in Intkey. For character images, these include the feature text box (Fig. 17d), the state text boxes (Fig. 17f), 'hotspots', and OK, Cancel, and Notes buttons.

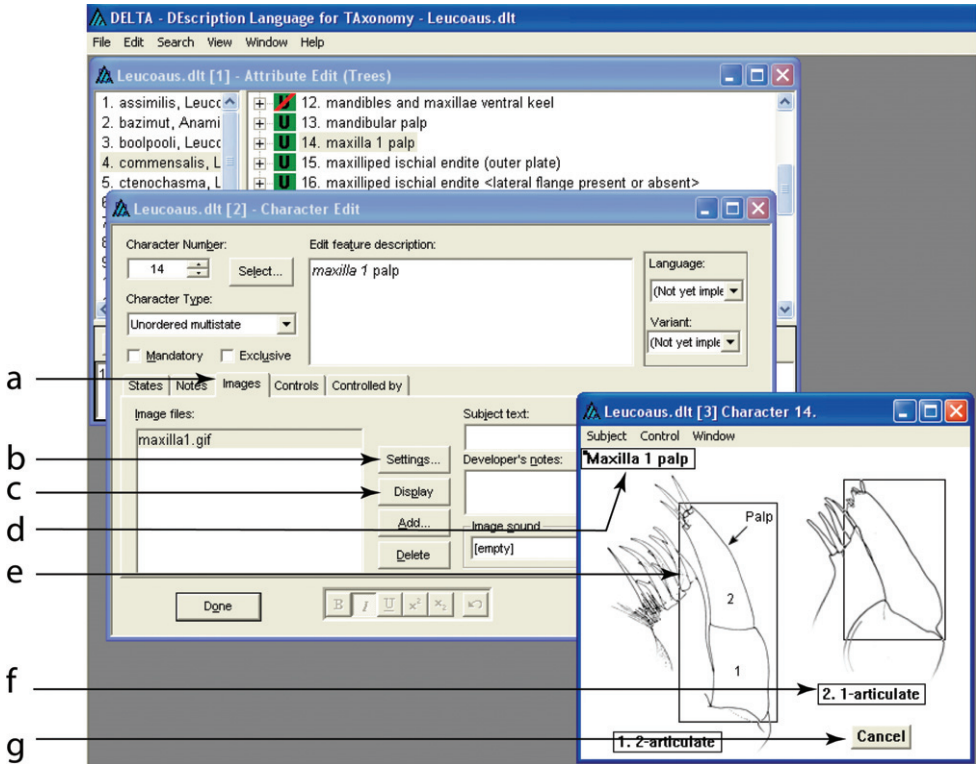


Figure 17a–g. Character state images. **a** the *Images* tab in the Character editor **b** *Settings* allow the selection of the path to the image directory and font details for the overlays **c** *Display* shows the selected image file in a window with the character and overlay boxes **d** feature text of the character **e** *Hotspot* selection box **f** character state overlay box **g** *Cancel* box. These overlays are used in Intkey.

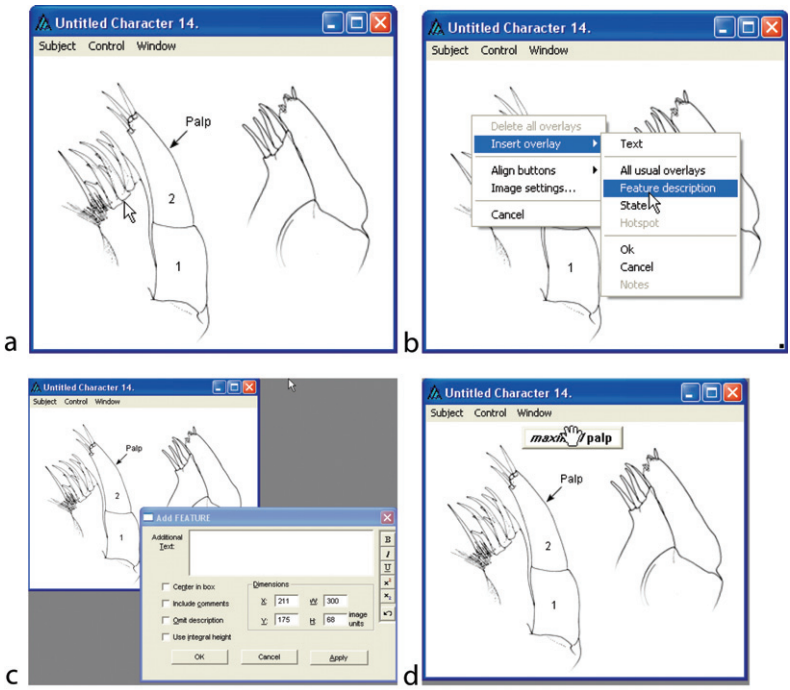


Figure 18a–d. Inserting an overlay with the feature text description on a character illustration.

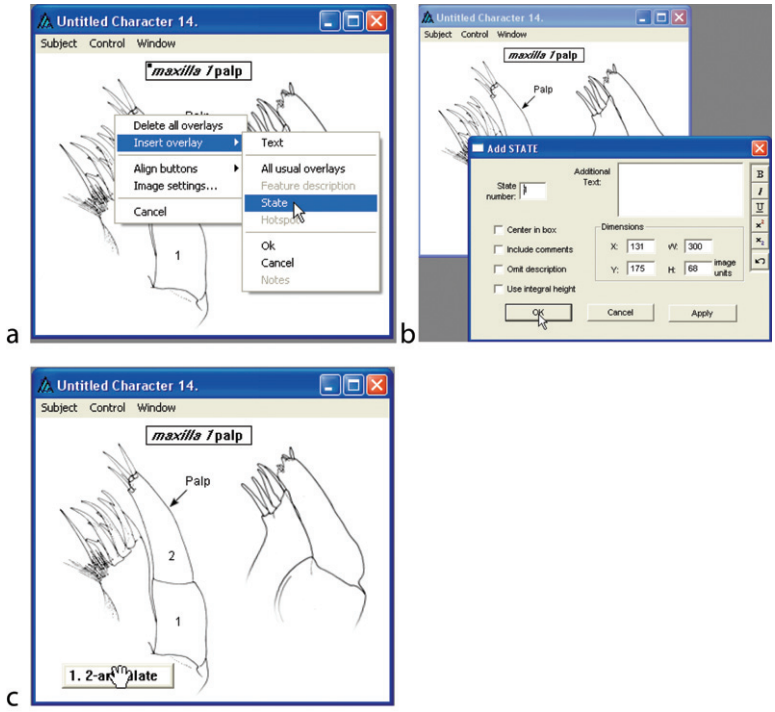


Figure 19a–c. Creating an overlay for character state illustrations.

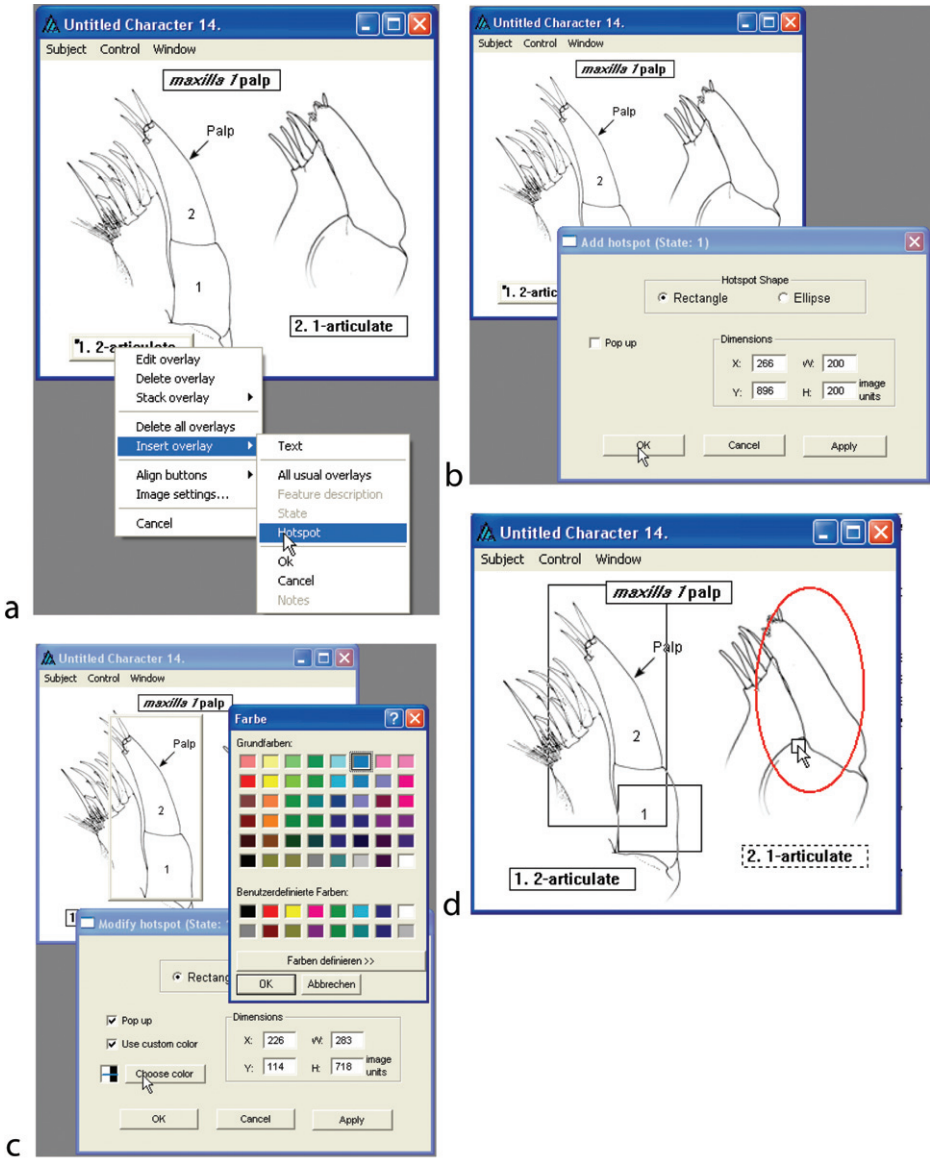


Figure 20a–d. Creating hotspots on a character image.

Hotspots are areas of the image that are linked to the state-text boxes. In Intkey, they assist in the identification and selection of character states. More than one hotspot may be associated with any state, and they may be invisible or 'pop-up'. When the cursor is moved over a hotspot, the associated state-text box becomes visibly active, and any associated pop-up hotspots become visible; the state may be selected by clicking on the hotspot. Conversely, when the cursor is moved over a state-text box, any associated pop-up hotspots become visible.

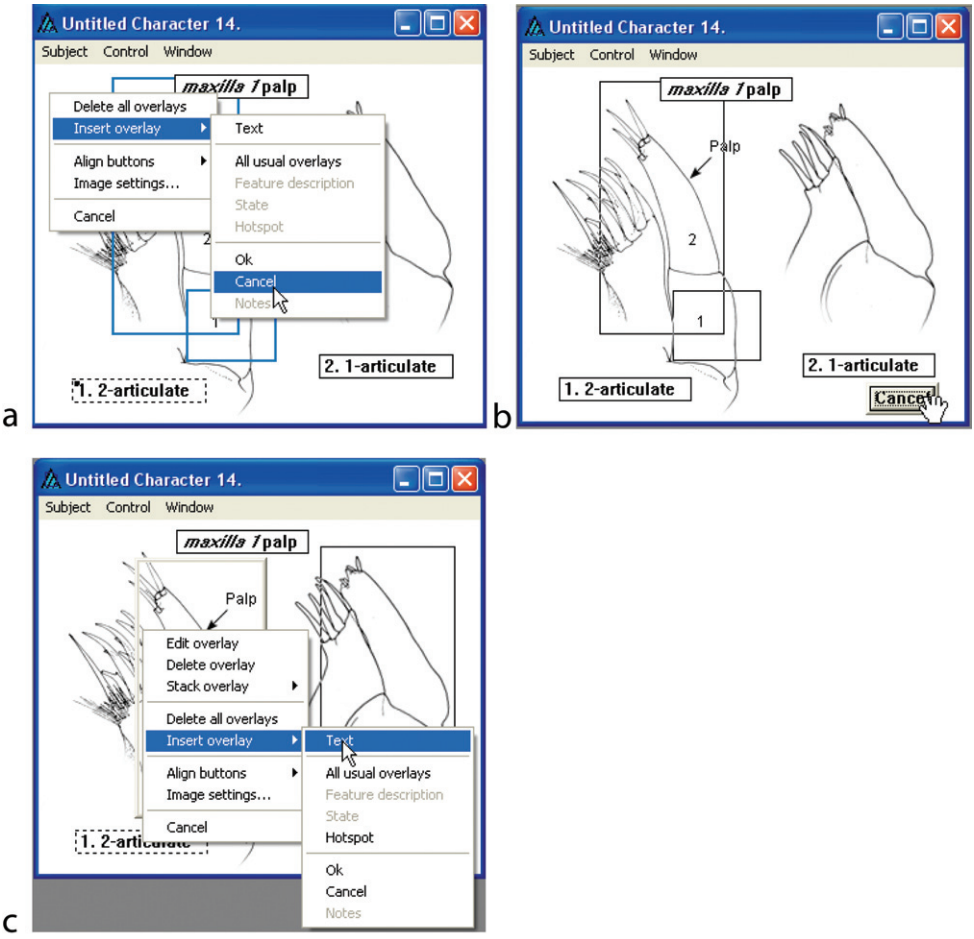


Figure 21. Inserting a *Cancel* button and additional text overlays on a character illustration.

It is easy to create these overlays. The screenshots (Figs 18–21) show how this is done.

1) Display the character image as described above.
2) Right click on the image, and select 'Insert overlay / All usual overlays'. A feature box, state boxes, and buttons are inserted, together with a single hotspot for each state.
3) Drag the overlays to appropriate positions on the image. Resize the text boxes and hotspots as necessary (by dragging a corner). Overlays may also be edited by right-clicking on the overlay and selecting 'Edit overlay'.
4) If required, add further overlays similarly. When adding additional hotspots, position the cursor over the relevant state box or an existing hotspot before right-clicking (otherwise the 'Hotspot' option will be grey in the context menu).

The feature box should normally be positioned at the top left of the image, and the state boxes under the relevant parts of the image. Buttons may be positioned anywhere that there is room for them, but it makes it easier for the end user if they are positioned consistently in every illustration; if necessary, add blank space to the image.

Normally, it is sufficient to have a single invisible hotspot for each state (as inserted by the 'All usual overlays' option). The hotspot should cover all of the area of the state illustration, so that the Intkey user can click anywhere on that illustration to select the state. If the state illustrations are crowded and somewhat intermingled, it may be necessary to place the state boxes in non-standard positions, and a pop-up hotspot can be used to allow the Intkey user to easily find the corresponding area of the image. Also, smaller pop-up hotspots can be used to pinpoint the most relevant part of a state illustration. For an example, see the illustration for 'culm nodes' in the DELTA sample data.

Outputs using the Action sets

An important part of every database program is its output function. This is available in the DELTA Editor in the so-called *Action sets* which can be called up from the drop-down menu *View/Action sets*.

The Action sets consist of a number of text files, used for standard output tasks. In each text file there is a list of directives (= commands) that are read in a sequence when the program Confor is run (which is usually done automatically from the DELTA Editor via the Run command, see later). Experienced users could write their own directive

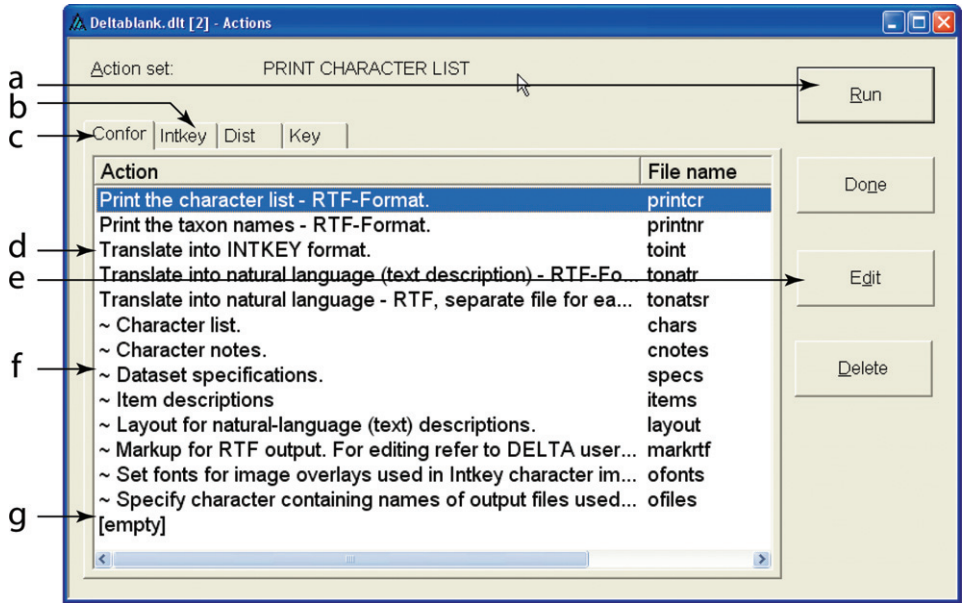


Figure 22a–g. The *Confor* dialog tab in Action sets.

file and select from over 80 directives. Due to this high number of commands DELTA is a very flexible system, but the Action sets often bewilder beginners.

Thus we have stripped down the Action sets of our Deltablank.dlt database by omitting the directives files for outputting the character list and natural-language descriptions as HTML (for Web publication), and for translating into formats for producing conventional keys, distance matrices, and cladistic trees (Nexus). We have also annotated each command in the Action sets where changes can be made by the user.

Here are a few things to keep in mind when using the directive files:

1) The sequence of directives within a directive file should not be changed. Certain directives need to keep a certain position in the sequence of the commands.
2) Each command consists of an asterisk and the directive text. All directives are listed in the user guide *uguide.doc* (see in folder c:\delta\doc, also available as html and pdf versions at http://delta-intkey.com/www/uguide.htm). Some commands have additions such as file or folder names. Many are followed by numbers which represent the taxa or character numbers used in the DELTA Editor. These numbers in the Action sets are automatically updated when changes are made in the DELTA Editor (e.g. if the position of a taxon in the taxon window is being changed and the taxon numbering is changed by that). Others should be edited directly in the Action set (e.g. linking characters). How this is done is explained in the User's Guide to the DELTA System.

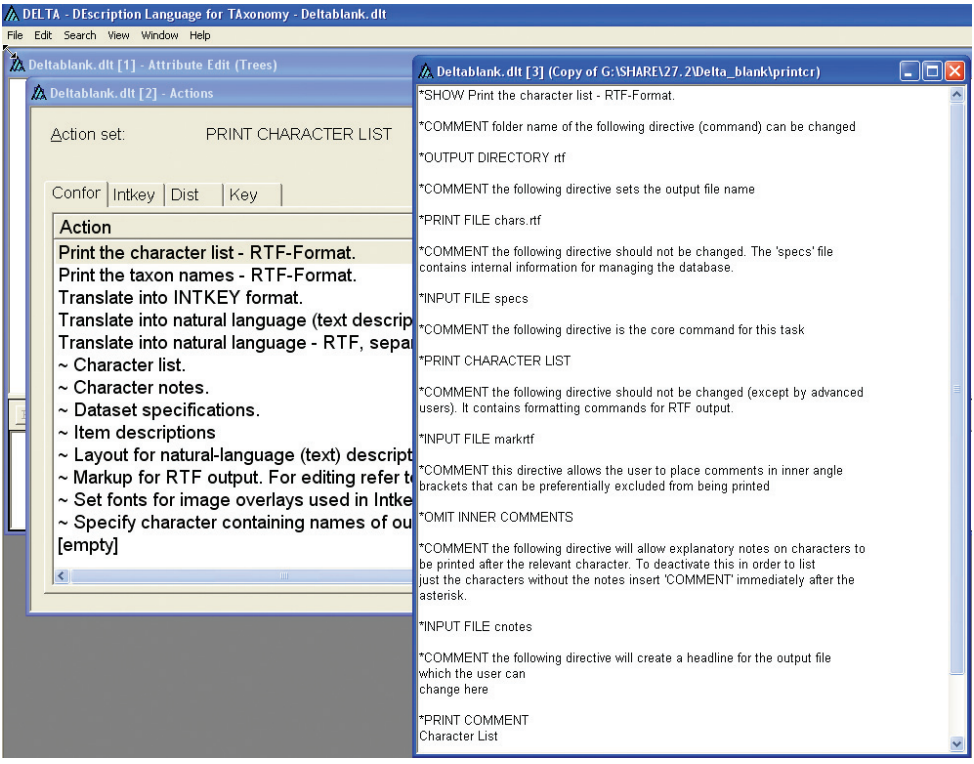


Figure 23. The *Print the character list* (*printcr*) directive file was opened by pushing the *Edit* button.

3) Commands can be switched off by inserting '*COMMENT* ' immediately after the * and prior to the command text (or switched on by omitting '*COMMENT* ' respectively).

Originally, DELTA data were edited as text, first on punched cards and later as computer files. The programs for processing these data ran successively on mainframes, minicomputers, and personal computers running MS-DOS and MS Windows. Eventually, the DELTA Editor was developed to maintain the DELTA data. It was intended that the Editor would also process the data, but the DELTA project was terminated before this could be achieved. Thus, the processing must still be done by the older programs. Before this can be done, the binary data used internally by the Editor must be exported as text files.

Exporting the data may be done directly by using File/Export directives from the main menu. Also, when an action set is run, the program asks whether the user wishes to export the DELTA text files, and warns that this should be done if the data have been changed since they were last exported (see Figure 24). This is a quick process so it is best to play safe by always doing this. After editing, it's advisable to check the data by running the important action sets, and to back up the exported text files as well as the .dlt binary file used by the Editor to a new location (i.e. don't overwrite previous backups). A convenient method is to use a zipfile with a name based on the current date.

Figure 22 shows the Action sets window which was selected from the dropdown menu *View*. Two of four tabs (Fig. 22b–c), *Confor* and *Intkey*, are described below.

The central dialog box under the *Confor* tab contains our selection of the essential directive files. Those marked with the tilde '~' (Fig. 22f) cannot be run (but only edited), they contain information that is used by the five output Action sets (Fig. 22d) at the beginning of the list. There is a position called *[empty]*, which can be used to create a new directive file. *File name* in the Confor dialog box is the name of the textfile which is created automatically in the working folder during the export. It is also used as a short name for the output function.

Two buttons on the right side are very important: *Edit* (Fig. 22e) and *Run* (Fig. 22a). *Edit* (Fig. 22e) opens the selected directive file (Fig. 23) and allows modifications to the file. *Run* (Fig. 22a) allows the execution of directive files.

The export procedure

For example, after the user has made changes to the directive file *printcr* (creating a text output of the characters used in the DELTA Editor), it is run by clicking on the *Run* button. A selection box (Fig. 24) asks the user, if an export of directive files should be carried out. When in doubt we would suggest to export. After clicking on the *Yes* button another window pops up (Fig. 25), the user does not change anything here but pushes the *Ok* button. The next window informs the user about the export procedure (Fig. 26), which is closed by clicking the *Done* button. The final window is the report from Confor on the Run procedure (Fig. 27). The window informs the user of a *Normal termination* (Fig. 27a)

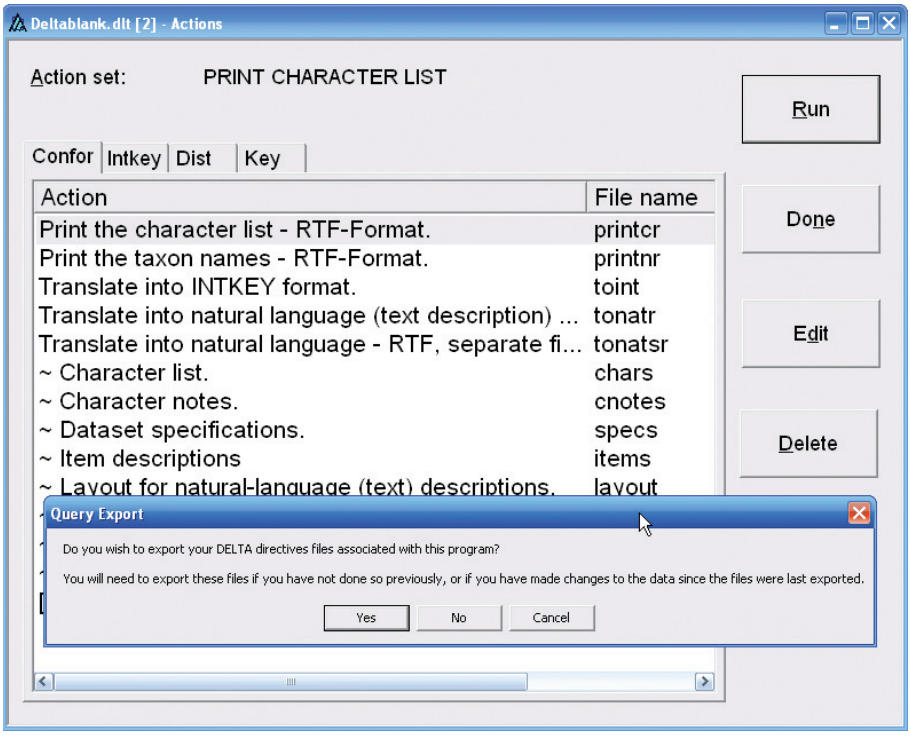


Figure 24. When running a directive file the program asks for an export of the directive files.

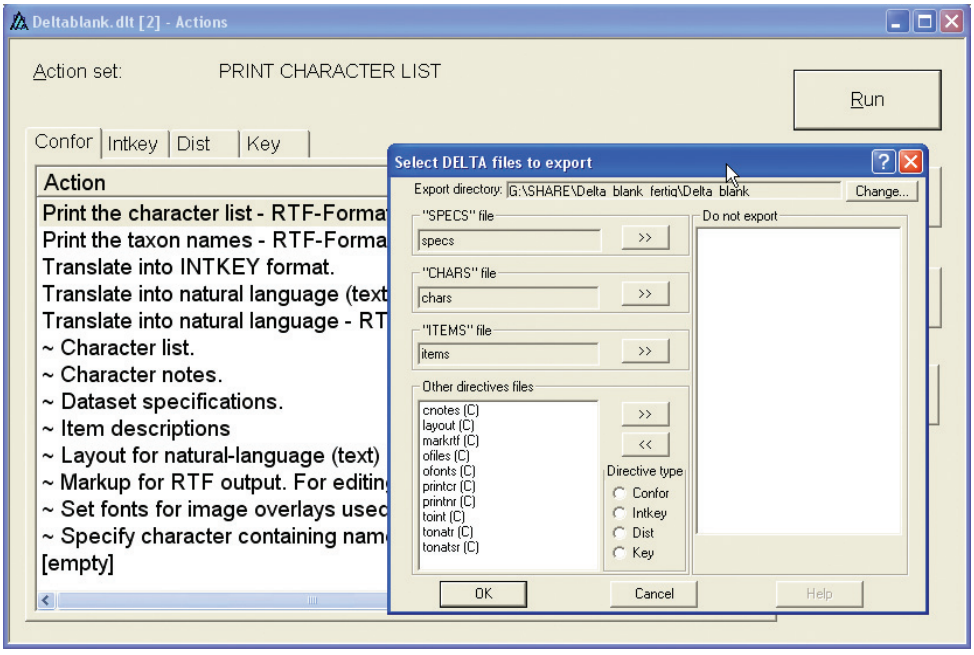


Figure 25. Confor in the DELTA Editor asks the user to make a selection of directive files being exported.

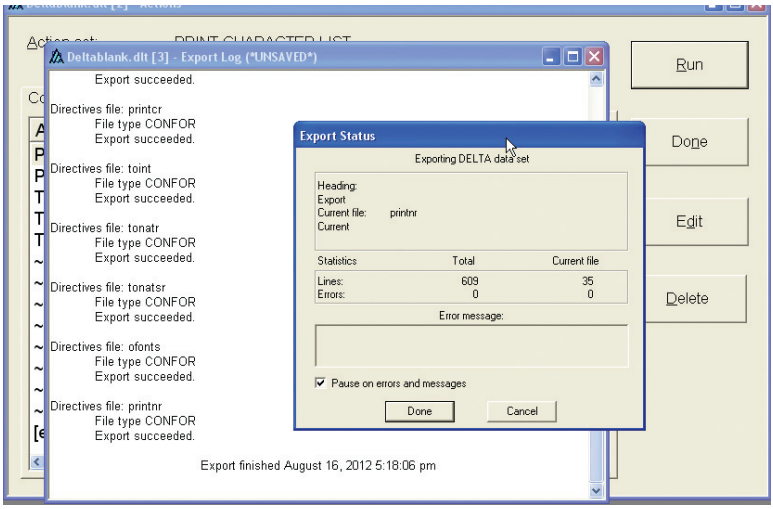


Figure 26. Messages about the course of the export procedures connected to running a directive file.

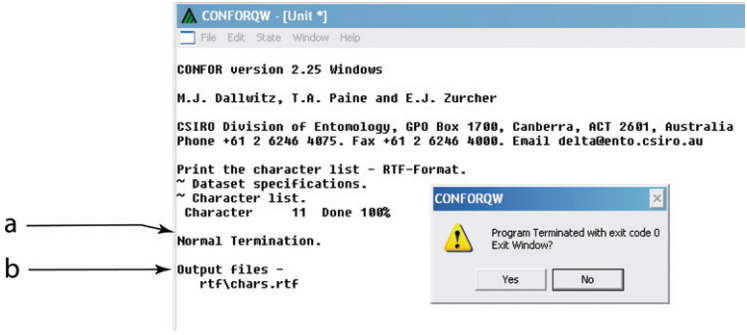


Figure 27a–b. Information window of Confor informing the user of a successful output: **a** it was terminated correctly and **b** the path shows were the output files are located.

(if this is not the case, it gives an indication of the problem) and where the output is stored (Fig. 27b): in the folder *.\rtf* of the working directory, saved under the file name *chars.rtf*.

Action sets used in the simplified template database *Deltablank.dlt* (see Fig. 22)

printcr

The directive file *printcr* (Fig. 29) produces a list of characters used in a DELTA database, including explaining comments for each character, as a file in rtf-format (Rich

```
2  *SHOW Print the character list - RTF-Format.

   *COMMENT folder name of the following directive (command) can be changed

5  *OUTPUT DIRECTORY rtf

   *COMMENT the following directive sets the output file name

5  *PRINT FILE chars.rtf

   *COMMENT the following directive should not be changed. The 'specs' file
   contains internal information for managing the database.

3  *INPUT FILE specs

   *COMMENT the following directive is the core command for this task

1  *PRINT CHARACTER LIST

   *COMMENT the following directive should not be changed (except by advanced
   users). It contains formatting commands for RTF output.

4  *INPUT FILE markrtf

   *COMMENT this directive allows the user to place comments in inner angle
   brackets that can be preferentially excluded from being printed

2  *OMIT INNER COMMENTS

   *COMMENT the following directive will allow explanatory notes on characters to
   be printed after the relevant character. To deactivate this in order to list
   just the characters without the notes insert 'COMMENT' immediately after the
   asterisk.

3  *INPUT FILE cnotes

   *COMMENT the following directive will create a headline for the output file
   which the user can change here

2  *PRINT COMMENT
   Character List

   *COMMENT the following directive should not be changed

3  *INPUT FILE chars
```

Figure 28. Directive file *printcr* for the output of characters used in a DELTA database. The numbers to the left of the directive file indicate different classes of directives (see in Table 1).

Table 1. Classification of commands used in the *printcr* directive file. Reference numbers used in Fig. 28 are given in the header of the table.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
PRINT CHARACTER LIST	SHOW	INPUT FILE specs	INPUT FILE markrtf	OUTPUT DIRECTORY rtf
	OMIT INNER COMMENTS	INPUT FILE cnotes		PRINT FILE chars. rtf
	PRINT COMMENT	INPUT FILE chars		

Text Format). The file name is somewhat misleading, this directive file does not output to a printer but generates a file.

printnr

The directive file *printnr* (Fig. 29) produces a list of taxon names used in a DELTA database as a file in rtf-format (Rich Text Format). The file name is somewhat misleading, this directive file does not output to a printer but generates a file.

```
(2) *SHOW Print the taxon names - RTF-Format.
    *COMMENT folder name of the following directive (command) can be changed
(5) *OUTPUT DIRECTORY rtf
    *COMMENT the following directive sets the output file name
(5) *PRINT FILE names.rtf
    *COMMENT the following directive should not be changed. The 'specs' file
    contains internal information for managing the database.
(3) *INPUT FILE specs
    *COMMENT the following directive is the core command for this task
(1) *PRINT ITEM NAMES
    *COMMENT the following directive should not be changed (except by advanced
    users). It contains formatting commands for RTF output.
(4) *INPUT FILE markrtf
    *COMMENT the following directive will create a headline for the output file
    which the user can
    change here
(2) *PRINT COMMENT
    Item Names
    *COMMENT the following directive should not be changed
(3) *INPUT FILE items
```

Figure 29. Directive file *printnr* for the output of taxon names used in a DELTA database. The numbers to the left of the directive file indicate different classes of directives (see in Table 2).

Table 2. Classification of commands used in the *printnr* directive file. Reference numbers used in Fig. 29 are given in the header of the table.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
PRINT ITEM NAMES	SHOW	INPUT FILE specs	INPUT FILE markrtf	OUTPUT DIRECTORY rtf
	PRINT COMMENT	INPUT FILE items		PRINT FILE names. rtf

toint

Running the directive file *toint* (Fig. 30) is the prerequisite for using the data from the DELTA Editor in Intkey, the interactive key program. *Toint* will process all relevant files needed to produce the data files for Intkey. It also includes commands which exclude characters from being used in Intkey, for defining the reliability of characters (their "weight" in the identification procedure) and some layout directives.

tonatr

This directive file is used for producing output from the DELTA database into a natural language description in the rtf-format (Rich Text Format).

tonatsr

This directive file is used for producing an output from the DELTA database into natural language descriptions in the rtf-format (Rich Text Format), but in contrast to *tonatr* it generates a separate file for each taxon. These individual descriptions are also

Table 3. Classification of commands used in the *toint* directive file. Reference numbers used in Fig. 30 are given in the header of the table. The commands in bold and italics may require changes being made by the user.

(1) Core command	**(2) Additional commands**	**(3) Input files**	**(4) Input layout files**	**(5) Output files**
TRANSLATE INTO INTKEY FORMAT	SHOW	INPUT FILE specs		INTKEY OUTPUT FILE ichars
	HEADING	INPUT FILE cnotes		INTKEY OUTPUT FILE iitems
	OMIT INNER COMMENTS	INPUT FILE cimages		
	CHARACTER FOR SYNONOMY	INPUT FILE timages		
	EXLUDE CHARACTERS	***INPUT FILE ofiles***		
	CHARACTER RELIABILITIES	INPUT FILE chars		
	NEW PARAGRAPHS AT CHARACTERS	INPUT FILE items		
	ITEM SUBHEADINGS			

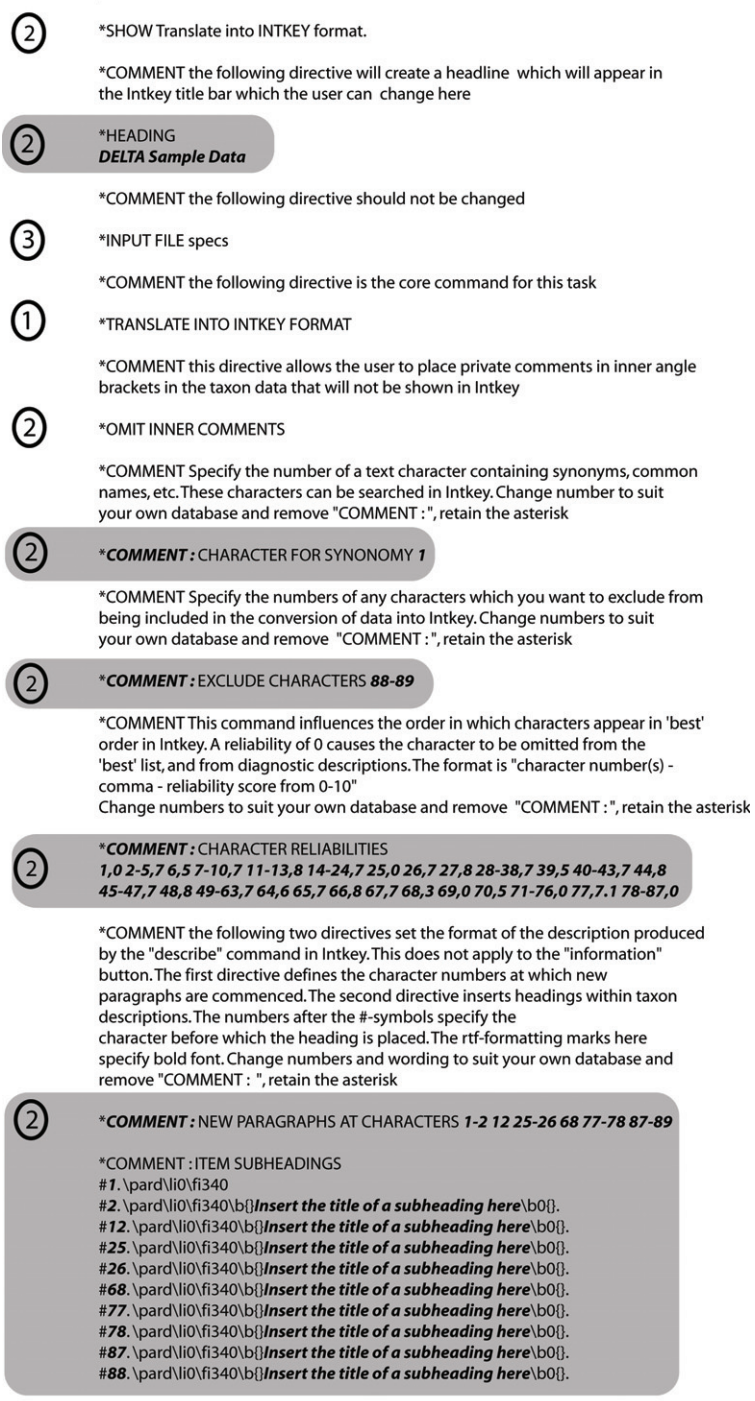


Figure 30. Directive file *toint* for generating the files needed for Intkey. The numbers to the left of the directive file indicate different classes of directives (see Table 3). The grey ovals show commands where input of the user might be required. (continued on next page)

*COMMENT the following directive will allow explanatory notes on characters (if used) to be available in appropriate context within Intkey. This directive should not be changed.

*INPUT FILE cnotes

*COMMENT the following directive should not be changed. It makes images of characters (if used) available in Intkey

*INPUT FILE cimages

*COMMENT the following directive should not be changed. It makes images of taxa (if used) available in Intkey

*INPUT FILE timages

*COMMENT If you want Intkey to link via the "information" button to descriptions generated by 'tonatsr', activate (by removing "COMMENT : ", retain the asterisk) the following directive. Before running this file (toint), you must modify 'ofiles', and make additions to your data as described in 'ofiles'.

***COMMENT :** INPUT FILE ofiles

*COMMENT the following directive should not be changed. Specifies the name of the output file for characters in Intkey format.

*INTKEY OUTPUT FILE ichars

*COMMENT the following directive should not be changed. Specifies the name of the characters file created by the DELTA Editor for conversion into Intkey format.

*INPUT FILE chars

*COMMENT the following directive should not be changed. Specifies the name of the output file for taxa in Intkey format.

*INTKEY OUTPUT FILE iitems.

*COMMENT the following directive should not be changed. Specifies the name of the taxa (items) file created by the DELTA Editor for conversion into Intkey format.

*INPUT FILE items

Table 4. Classification of commands used in the *tonatr* directive file. Reference numbers used in Fig. 31 are given in the header of the table. The commands in bold and italics may require changes being made by the user.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
TRANSLATE INTO NATURAL LANGUAGE	SHOW	INPUT FILE specs	INPUT FILE markrtf	OUTPUT DIRECTORY rtf
	PRINT COMMENT	INPUT FILE chars	INPUT FILE layout*	PRINT FILE descrip.rtf
	TRANSLATE IMPLICIT VALUES	INPUT FILE items		

* The layout file needs to be adapted before *tonatr* can be run successfully.

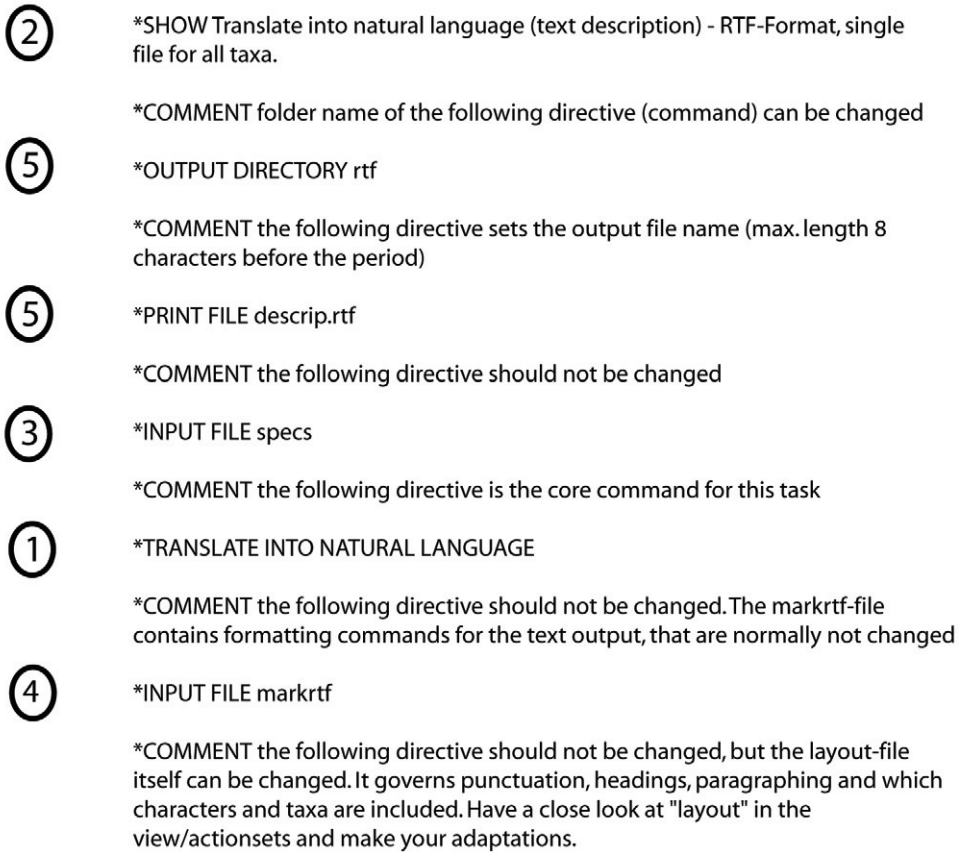
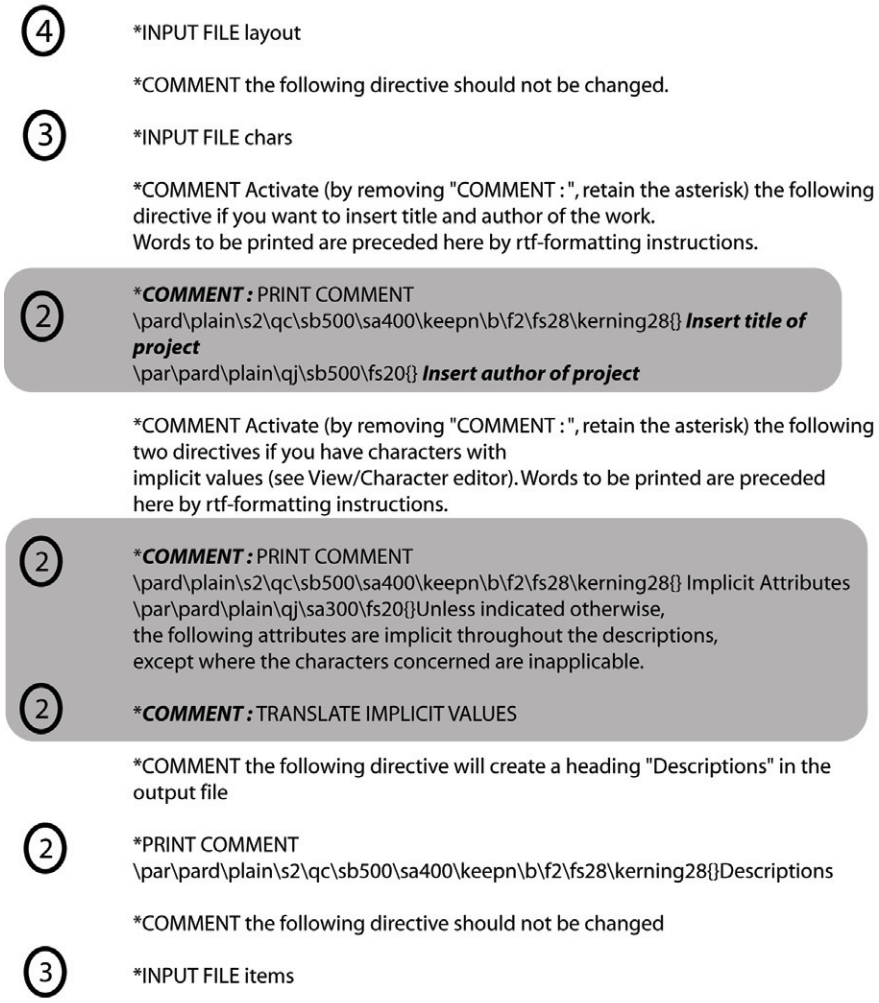


Figure 31. The *tonatr* directive file contains the commands for producing natural language descriptions. The numbers to the left of the directive file indicate different classes of directives (see in Table 4). The grey ovals show commands where input of the user might be required. (continued on next page).



used by Intkey to display a formatted text description in Intkey for each individual species when the blue Information button is activated (explained in detail in the section on Intkey).

chars

The *chars* file contains the characters and their states in DELTA format. This file cannot be changed or edited directly. All changes made within the DELTA Editor via the Character Editor screen to any of the characters, their sequence in the list and the character states, are updated automatically by the DELTA Editor.

Table 5. Classification of commands used in the *tonatsr* directive file. Reference numbers used in Fig. 32 are given in the header of the table. The commands in bold and italics may require changes being made by the user.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
TRANSLATE INTO NATURAL LANGUAGE	SHOW	INPUT FILE specs	INPUT FILE markrtf	OUTPUT DIRECTORY info
	PRINT COMMENT	INPUT FILE ofiles*	INPUT FILE layout*	PRINT FILE implicit.rtf
	TRANSLATE IMPLICIT VALUES	INPUT FILE chars		
		INPUT FILE items		

* The *layout* and *ofiles* directive files need to be adapted before *tonatsr* can be run successfully.

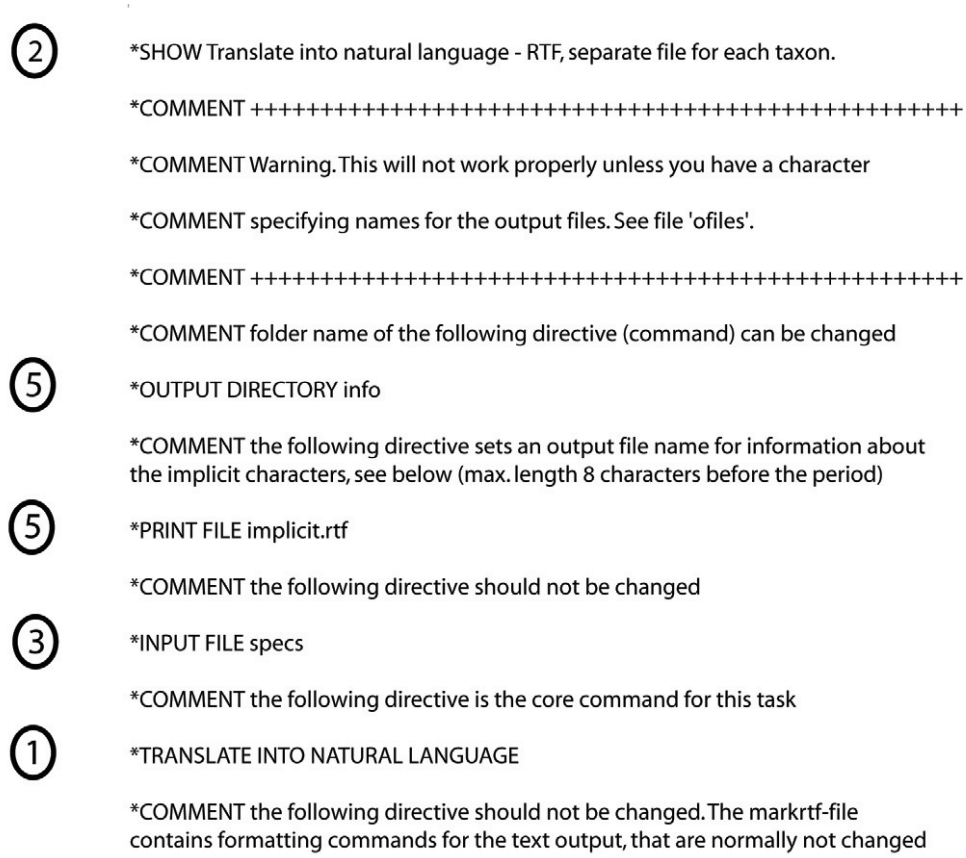
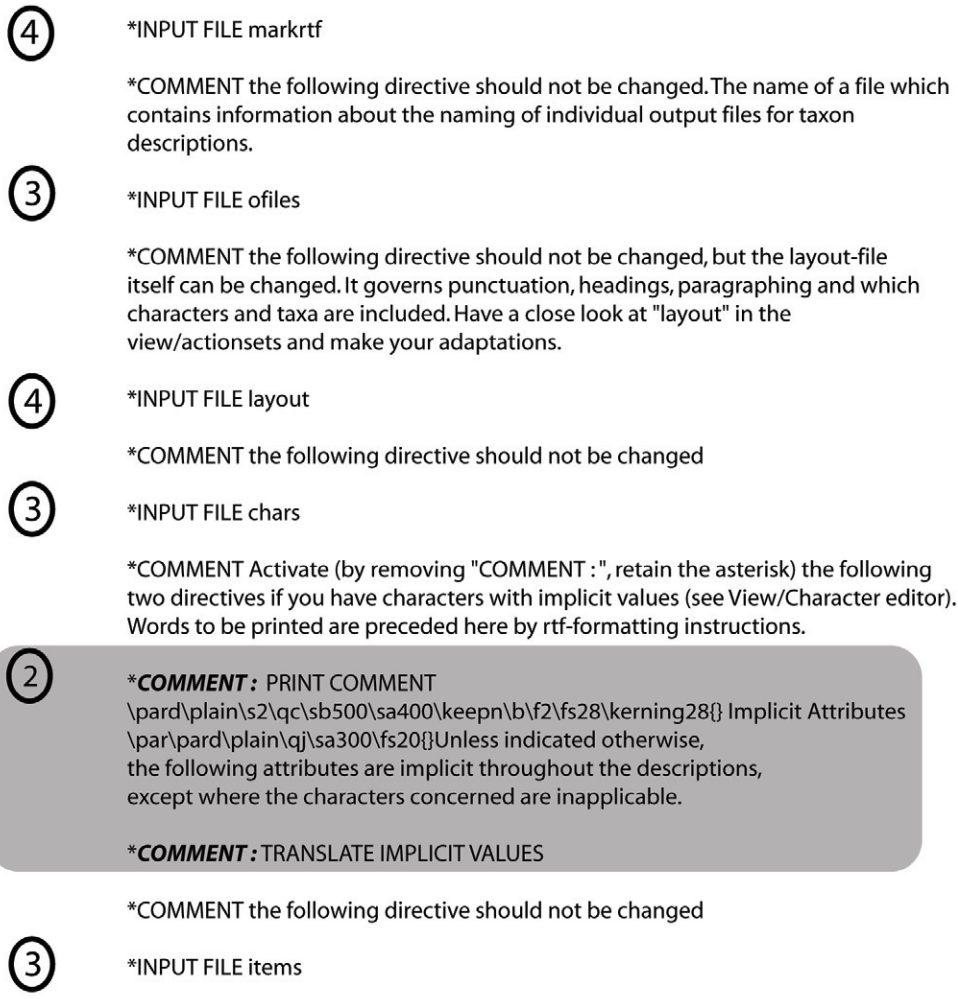


Figure 32. The tonatsr directive file contains the commands for producing natural language descriptions as separate files for each taxon. The numbers to the left of the directive file indicate different classes of directives (see in Table 5). The grey oval shows commands where input of the user might be required. (continued on next page).



cnotes and cimages

Cnotes and *cimages* are two files that do not need to be changed. They are generated automatically by the DELTA Editor when notes and images are being made in the character editor (see there under the tabs *Notes* and *Images*). These data are used in Intkey to explain characters in text form (*cnotes*) and by illustration (*cimages*).

specs

The *specs* file contains information that is used by DELTA to manage the database, such as numbers of characters and taxa, the type of each character and their number of

states, implicit values and controlling characters. This file cannot be changed directly, but DELTA will do this job for us in the background.

items

Items is another file that is updated automatically by the DELTA Editor in the background. It contains the DELTA-format description of each item (taxon).

layout

The *layout* file is used by *tonatr* and *tonatsr* to format natural language output. For example in this file the user can define where paragraphs should appear in the text output, which characters should be linked together to form a sentence and where bold subheadings should appear. It is supported by the *markrtf* file which contains the commands for rtf formatting.

Table 6. Classification of commands used in the *layout* directive file. Reference numbers used in Fig. 33 are given in the header of the table. The commands in bold and italics may require changes being made by the user.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
	REPLACE ANGLE BRACKETS			
	OMIT CHARACTER NUMBERS			
	OMIT INNER COMMENTS			
	OMIT INAPPLICABLES			
	CHARACTER FOR TAXON IMAGES			
	EXCLUDE CHARACTERS			
	NEW PARAGRAPHS AT CHARACTERS			
	LINK CHARACTERS			
	ITEM SUBHEADINGS			

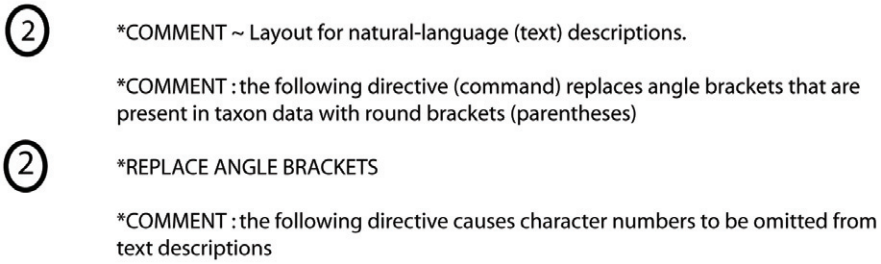


Figure 33. The *layout* directive file contains the formatting commands for producing natural language descriptions. The numbers to the left of the directive file indicate different classes of directives (see Table 6). The grey ovals show commands where input of the user might be required. (continued on next page).

2 *OMIT CHARACTER NUMBERS

*COMMENT the following directive allows the user to place comments in inner angle brackets that can be preferentially excluded from being printed

2 *OMIT INNER COMMENTS

*COMMENT : the following directive suppresses the printing of inapplicable characters in text descriptions.

2 *OMIT INAPPLICABLES

*COMMENT : in HTML-format descriptions the following directive specifies a location (by means of the character number) in the description where image links will appear .
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'). Activate directive by removing "COMMENT : ", retaining the asterisk

2 ***COMMENT :** CHARACTER FOR TAXON IMAGES **88***

*COMMENT : the following directive allows characters to be excluded from a conversion of data to descriptions by listing their numbers here. Activate directive by removing "COMMENT : ", retaining the asterisk

2 ***COMMENT :** EXCLUDE CHARACTER **89***

*COMMENT the following directive defines the character numbers at which new paragraphs are commenced in descriptions. Activate directive by removing "COMMENT : ", retaining the asterisk

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

*COMMENT Group characters into sentences by means of the character numbers. Activate directive by removing "COMMENT : ", retaining the asterisk

2 ***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 taxon descriptions. The numbers after the #-symbols specify the character before which the heading is placed. The rtf-formatting marks here specify bold font. Activate directive by removing "COMMENT : ", retaining the asterisk

2 ***COMMENT :** ITEM SUBHEADINGS
#2. \b{}***Insert the title of a subheading here***\b0{}.
#12. \b{}***Insert the title of a subheading here***\b0{}.
#25. \b{}***Insert the title of a subheading here***\b0{}.
#26. \b{}***Insert the title of a subheading here***\b0{}.
#68. \b{}***Insert the title of a subheading here***\b0{}.
#77. \b{}***Insert the title of a subheading here***\b0{}.
#78. \b{}***Insert the title of a subheading here***\b0{}.
#87. \b{}***Insert the title of a subheading here***\b0{}.
#88. \b{}***Insert the title of a subheading here***\b0{}.

markrtf

The *markrtf* file contains typesetting commands for the rtf outputs for the natural language descriptions and also for producing the output file for a taxon and character list (*printcr* and *printnr*). Changing *markrtf* requires an understanding of its actions, but in most cases it should be fine as it is.

ofonts

Ofonts contains settings for the overlays on illustrations automatically transferred by DELTA into this file. These can be selected in *Settings* when assigning pictures to characters and taxa with the Taxon editor and the Character editor.

For experienced users: *ofiles*

The *ofiles* are very useful to automatically assign file names for files generated with *tonatsr* (natural language descriptions that are output as separate files). These natural language descriptions are used in Intkey to show a full description for each taxon. For using the *ofiles* we:

1) create a (mandatory) text character (eg. <prefix for ofile names>) with the Character editor of the DELTA Editor,
2) insert a unique text string for each taxon not longer than 8 characters and
3) put in the number of this text character into *ofiles* behind the command CHARACTER FOR OUTPUT FILES and activate this command by erasing '*COMMENT:*then
4) activate in the *toint* and *tonatsr* directive file the command '*INPUT FILE ofiles'* by erasing the '*COMMENT:*'
5) When *tonatsr* and then *toint* are run, *ofiles* will be read in: *tonatsr* will create the individual taxon descriptions, *ofiles* will generate the file names, as defined for each taxon by the user in the mandatory textfield, and *toint* will connect these files with the taxon names in Intkey.

empty

There is a line in the Action sets called *empty* (Fig. 22g). This is important for writing your own directive files or importing directive files from other databases by simply copying the text in the window that opens after clicking the *Edit* button (Fig. 22e).

Table 7. Classification of commands used in the *ofiles* directive file. Reference numbers used in Fig. 34 are given in the header of the table. The command in bold and italics may require changes being made by the user.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
	CHARACTER FOR OUTPUT FILES			
	SUBJECT FOR OUTPUT FILES			

```
*COMMENT ~ Specify character containing names of output files used in Intkey.

*COMMENT the following directive specifies a text character, by means of the
number after "CHARACTER FOR OUTPUT FILES", containing an abbreviation of the
taxon name. The abbreviation can have a maximum of 8 letters and must not contain
spaces. This abbreviation is used in 'tonatsr' (for RTF output) and 'tonath'
(for HTML output) to generate the name of a file which is to contain the
individual natural-language description of the taxon for display in Intkey. The
abbreviation is also used in 'toint' to provide links in Intkey to individual descriptions
of taxa (which are generated by 'tonatsr'). The character should be
specified 'mandatory' in the 'View/Character editor'.
Example:
---This character number has been set by the database author to contain the
taxon name abbreviation: 89.
---Full taxon name as it occurs in the database: Festuca baffinensis.
Abbreviation chosen by the database author for this taxon name: festbaf (to
be placed in character 89)
---Output file name: festbaf.rtf (for RTF output) or festbaf.htm (if HTML output
is required).

2  *COMMENT : CHARACTER FOR OUTPUT FILES 89

*COMMENT  To set the text which will appear in Intkey, when the blue Information
button is selected.

2  *SUBJECT FOR OUTPUT FILES Full description
```

Figure 34. The *ofiles* directive file contains data for the automatic naming of files created with *tonatsr*. The numbers to left of the directive file indicate different classes of directives (see in Table 7). The grey oval shows command where input of the user might be required.

Intkey tab with inkey.ink

Just right of the *Confor* tab in the Action sets window there is the *Intkey* tab (Fig. 22b). Under this tab the user finds the Intkey.ink and toolbar.inp (a file for a user defined toolbar, see in comments in Intkey.ink for details). When *Inkey.ink* is run Intkey will be started using the commands specified in *Intkey.ink*.

In Intkey.ink there are three useful commands, when dealing with a large number of characters or taxa. With DEFINE CHARACTERS, DEFINE TAXA and DEFINE NAMES sets of characters or taxa can be grouped and called up with a keyword during identification.

```
*COMMENT Intkey initialization file

*COMMENT : This file specifies the Intkey data files and makes settings in
Intkey at the beginning of each session.

*COMMENT : The following directive (command) specifies the folder path which
will be searched for images that illustrate the characters and taxa. The default
is the current folder (the data folder, i.e. the folder which contains the database).
In this example Intkey will search for the images in the data folder and in a subfolder
called 'images'. If desired the name of this folder could be changed by the
user. It is also possible to cite an URL where the images are located on the
web.

2  *SET IMAGEPATH images

*COMMENT : The following directive specifies the folder path which will be
searched for files (e.g. individual taxon descriptions created by running 'tonatsr') available for
display when the blue Information button in Intkey is selected. The default is the current folder
(the data folder, i.e. the folder which contains the database). In thisexample Intkey will search
for the description files in the data folder and in a subfolder called 'info'. If desired the
name of this folder could be changed by the user.
It is also possible to cite an URL where the files are located on the web.

2  *SET INFOPATH info

*COMMENT : The following directive should not be changed. This sets the name of
the file containing taxon data in a format required by Intkey, produced
automatically by the 'toint' action set.

3  *FILE TAXA iitems

*COMMENT : The following directive should not be changed. This sets the name of
the file containing characters data in a format required by Intkey, produced
automatically by the 'toint' action set.

3  *FILE CHARACTERS ichars

*COMMENT : The user can define their own buttons for the additional toolbar of
Intkey by means of a set of button definitions. This command specifies the name
of the file containing the button definitions and causes it to be read (or
input) into Intkey. Intkey comes with standard inbuilt buttons which are usually
adequate. If the user wants to read in their own toolbar definitions, this
directive needs to be activated by  removing "COMMENT : ", retain the asterisk.

3  *COMMENT : FILE INPUT toolbar.inp

*COMMENT : The following directive needs not to be changed. This default value
normally gives adequate results. Works together with 'CHARACTER RELIABILITES' in
the 'toint' action set.

2  *SET RBASE 1.2

*COMMENT : The following directive determines whether unknown (uncoded)
characters are displayed in the output of the DESCRIBE command within Intkey.
The advanced user can turn this option on within Intkey in the pulldown menu
'Settings/Display/Unknowns ON/OFF'.
```

Figure 35. The *Intkey.ink* directive file specifies the Intkey data files and makes settings in Intkey at the beginning of each session. The numbers to the left of the directive file indicate different classes of directives (see in Table 8). The grey ovals show commands where input of the user might be required. (continued on next page).

*DISPLAY UNKNOWNS Off

*COMMENT : The following directive determines whether inapplicable characters are displayed in the output of the DESCRIBE command within Intkey. The advanced user can turn this option on within Intkey in the pulldown menu 'Settings/Display/Inapplicables ON/OFF'.

*DISPLAY INAPPLICABLES Off

*COMMENT : The following directives DEFINE CHARACTERS are examples of how to define Character Keywords, which are group names for sets of related characters. Once defined these sets enable easy reference to multiple characters in Intkey commands. The keywords should be enclosed in quotes. The numbers are character numbers (or ranges). Change text and numbers to suit your own database and remove "COMMENT : ", being sure to retain the asterisk.

*COMMENT : DEFINE CHARACTERS "***keyword 1***" ***1***

*COMMENT : DEFINE CHARACTERS "***keyword 2***" ***7-11***

*COMMENT : DEFINE CHARACTERS "***keyword 3***" ***86***

*COMMENT : The following directive is like DEFINE CHARACTERS except it defines sets of taxa. It is generally preferable to use taxon numbers as in the DEFINE TAXA directive rather than the alternative format DEFINE NAMES which uses taxon names, because position changes of taxa in the DELTA Editor are automatically updated. Change text and numbers to suit your own database and remove "COMMENT : ", being sure to retain the asterisk.

*COMMENT : DEFINE TAXA ***taxaset 7-8 10 11 14***

*COMMENT : The following directive is like DEFINE TAXA except it defines sets of taxa by means of names instead of numbers. Commas separate names and quotes must be used for names containing spaces (e.g. "Poa annua", "Zea mays"). Change text to suit your own database and remove "COMMENT : ", being sure to retain the asterisk.

*COMMENT : DEFINE NAMES ***taxaset Taxon1, Taxon2, Taxon3, Taxon4, Taxon5***

*COMMENT : The following directive is an optional alternative means to produce full description of the taxa to the method of pregenerating RTF descriptions by the 'tonatsr' actionset. The text that appears between the first set of quotes will be listed in the window that is opened when the blue Information button in Intkey is selected.

*DEFINE INFORMATION "Full description (generated by Intkey)" "describe ?S /c qall"

*COMMENT : The following directive produces the entry 'Diagnostic description' in the window that is opened when the blue Information button in Intkey is selected. When chosen, it leads to a diagnostic description being generated by Intkey. The parameter '?S' means the set of taxa currently selected in Intkey. The parameter 'none' means that no characters are preset and so the program produces a diagnostic description by considering all characters.

*DEFINE INFORMATION "Diagnostic description" "diagnose ?S none"

Table 8. Classification of commands used in the *Intkey.ink* directive file. Reference numbers used in Fig. 35 are given in the header of the table. The command in bold and italics may require changes being made by the user.

(1) Core command	(2) Additional commands	(3) Input files	(4) Input layout files	(5) Output files
	SET IMAGE PATH images	FILE TAXA iitems		
	SET INFOPATH info	FILE CHARACTERS ichars		
	SET BASE	***FILE INPUT toolbar.inp***		
	DISPLAY UNKNOWNS			
	DISPLAY INAPPLICABLES			
	DEFINE CHARACTERS			
	DEFINE TAXA			
	DEFINE NAMES			
	DEFINE INFORMATION			

Interactive keys - Intkey

Intkey is the information retrieval and interactive identification program of the DELTA package. It processes taxon and character related data from the DELTA Editor. Intkey is very powerful and the user can select from dozens of functions to access and use these data, for example to find differences and similarities between taxa or diagnosing them. The heart of Intkey is its interactive identification function. Unlike traditional dichotomous printed keys, the user is free to select any characters he wants to use during an interactive identification in Intkey.

In the course of the next pages we shall explain in short the user interface and give a detailed example of how to identify taxa interactively and how we construct dichotomous keys with Intkey (instead of using the *key* Action set to generate a key).

How to start Intkey

Intkey can be started in three ways:

1) from the DELTA Editor by running Intkey.ink from the Action sets tab *Intkey* (with an analogous export procedure described from the Confor tab).
2) in the database folder (using the file manager) by clicking on an Intkey initialisation file (recognisable by the file extension *.ink*)
3) in the DELTA folder (using the file manager) by opening the Intkey program (Intkey5.exe) and selecting an initialization file (a file with the extension *.ink*) or a title that has been saved before from the opening dialog.

Shortcuts to 2 and 3 can be placed elsewhere, e.g. on the Desktop.

We offer a sample database of Australian leucothoid amphipods for demonstration and testing for download here: http://download.naturkundemuseum-berlin.de/oliver.coleman/delta/AusLeucothoidae.zip

Interface

Main panes

Intkey has 4 main panes (Fig. 36): *Best Characters* in the upper left pane lists the available characters not in the natural order but in a sequence calculated by Intkey that allows fastest identification. The characters in the list appear first which tend to divide the taxa available for identification into almost equally large groups. The index used by Intkey for the *Best Characters* selection can be made visible by switching into advanced mode (*File/Advanced mode*).

The upper right pane contains the *RemainingTaxa*. During identification this lists becomes shorter and the contents of *Eliminated Taxa* in the lower right pane becomes longer. The lower left pane contains the *Used Characters* which were selected from the *Best Characters* list during identification.

Drop down menus

The user can select between two modes: the program starts in *Normal mode*, under the *File* drop down menu it can be switched into *Advanced mode*. When *Advanced mode* is selected the number of drop down menu entries is larger and there are two more buttons in the Characters toolbar. Normal mode is sufficient for most operations, particularly identification.

For experienced users: *Adapting Toolbars*

In Intkey there are 2 rows of buttons (right of *Best Characters* and *Remaining Taxa*, Fig. 36). A third row of user-defined buttons can be loaded by activating the command '*FILE INPUT toolbar.inp'* (by erasing COMMENT) in the Intkey.ink directive file. When Intkey.ink is run the definitions listed in the file toolbar.inp are then used for this database. In the toolbar.inp directive file (under the Intkey tab of Action sets in the DELTA Editor) the buttons and the allocated function can be modified. The user defined toolbar appears right below the drop down menus.

In our *Deltablank* database we included 5 useful buttons. Some buttons, like the first 3 at the left of the row, belong to the standard sets and are included with the Intkey program. Others, like the two red buttons, must be stored as graphic files in the bmp-format in the folder where Intkey.ink is located, normally the folder created for our database file (in our example *Deltablank.dlt*).

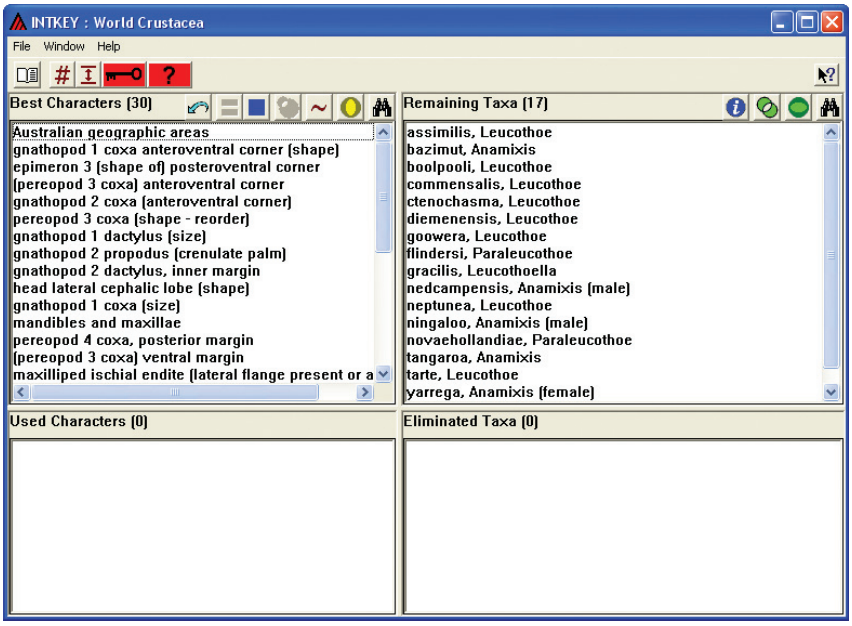


Figure 36. Sample file of Australian leucothoid amphipods in Intkey.

Buttons

We now explain all the buttons used in the sample database and Intkey files generated with the *Deltablank.dlt*.

User defined toolbar (definition of buttons and connected commands in the *toolbar.inp* directive file under the Intkey tab in the Action sets of the DELTA Editor) (Figs 37–41):

1) *Introduction and references* (Fig. 37): Clicking this buttons opens a dialog with background information related to the taxa appearing in the Intkey file. This selection menu is created by the file *contents.ind* which should be stored together with the connected rtf-files (file names not longer than 8 characters) in the folder where Intkey is located. The format of entries in this file is:

   '*Headline *FILE DISPLAY file_name.rtf* .
   In our example:
   *Introduction *FILE DISPLAY intro.rtf*
   *Checklist *FILE DISPLAY chcklst.rtf*
   *Museum Abbreviations *FILE DISPLAY MusAbbre.rtf*
   *Acknowledgements *FILE DISPLAY ack.rtf*
   *References *FILE DISPLAY refs.rtf*

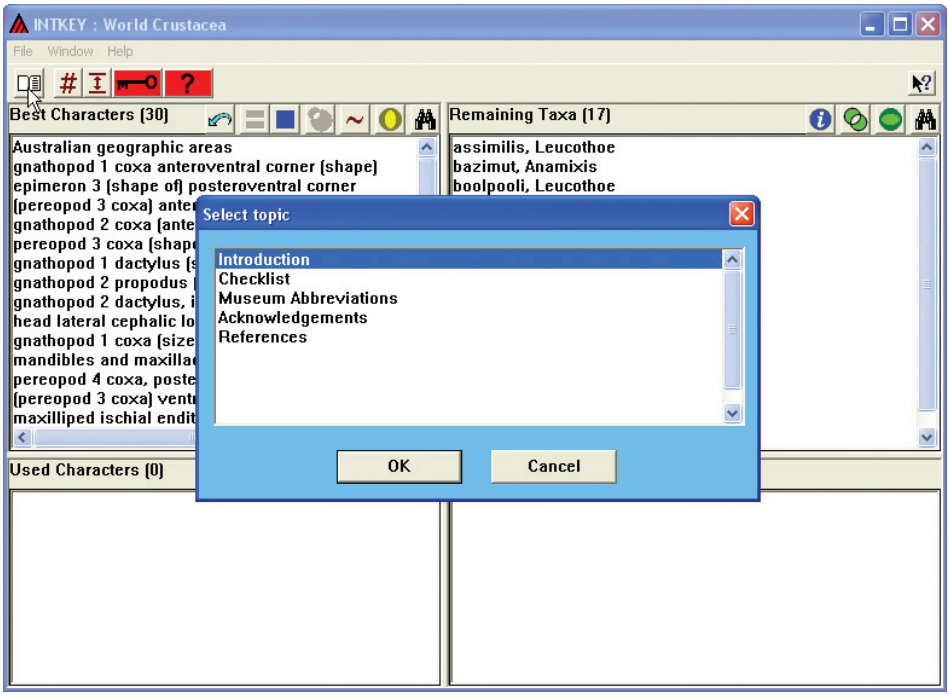


Figure 37. *Introduction and references* button.

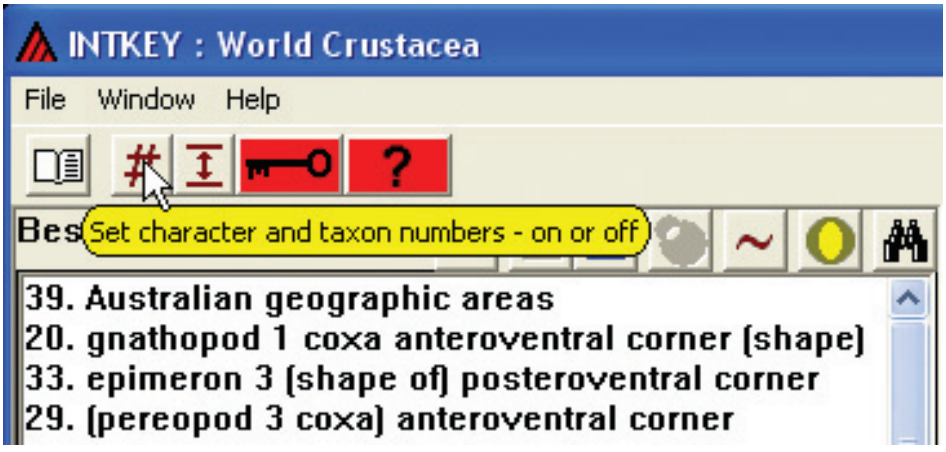


Figure 38. *Set character and taxon numbers* button.

2) *Set character and taxon numbers – on or off* (Fig. 38). Clicking on this button allows to switch the numbers preceding the characters and taxa on and off.

3) *Set diagnostic level* (Fig. 39). Clicking on this button allows changing the 'diagnostic level'. If a higher number is selected then more diagnostic features or combina-

tions of such features are listed when requesting a diagnosis description of a taxon (see explanation of Information button and Figs 53ff).

4) The *Identification* button (Fig. 40) changes Intkey into the identification mode where also unknown character states are allowed for the identification process. If a character is not recorded the program assumes in this mode that all character states of a character are possible.

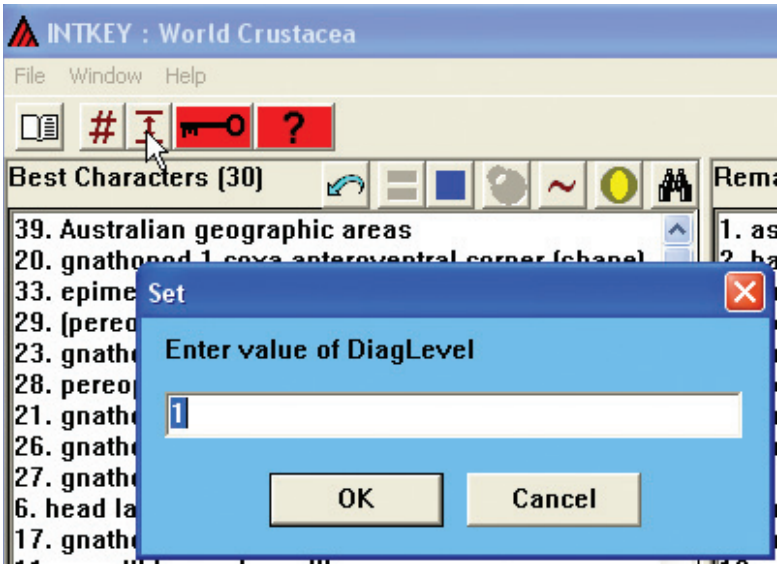


Figure 39. *Set diagnostic level* button.

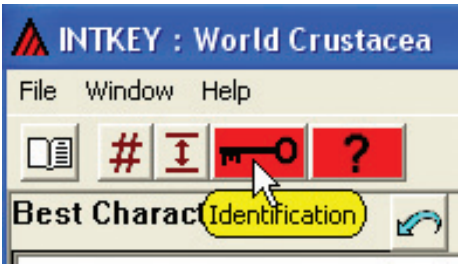


Figure 40. *Identification* button.

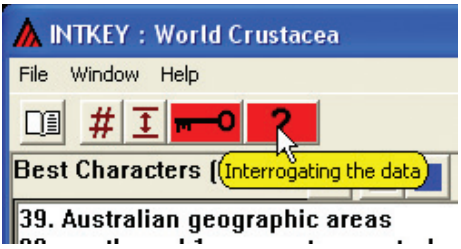


Figure 41. *Interrogating the data* button.

5) In contrast the *Interrogating the data* button (Fig. 41) switches into the mode where unknown character states are excluded from the information retrieval. Only the data that were actually recorded are used.

Best Characters toolbar (Figs 42–51):

1) *Restart identification or query* button (Fig. 42). Clicking on this button clears any used characters and restarts the session.

2) *Natural order* and *Best order* buttons (Figs 43–44). With these buttons it is possible to switch between the (natural) order of characters (Fig. 43), which is the order they have in the characters list in the DELTA Editor and the *Best* sequence, calculated by Intkey. Sorting the characters according to the *Best* index enables the user to split the group of items in *Remaining taxa* in about equally large groups when selecting a character from the top of the list (or with a *Best* index of about 1). In order to make the *Best* index visible the user should change into *Advanced mode* (Fig. 45). The *Best* index is shown in the left hand column in the *Best Characters* window as an index number smaller or larger than 1 (Fig. 46). In *Advanced mode* a third blue order icon appears: *Separate a given taxon order*. After selecting a taxon in the *Remaining Taxa* window a click on this button reorders the available characters according to how well they separate this taxon from the other remaining taxa.

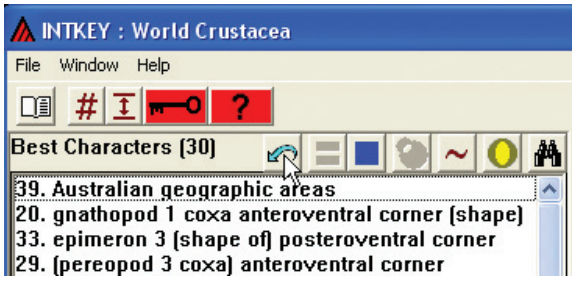


Figure 42. *Restart identification or query* button.

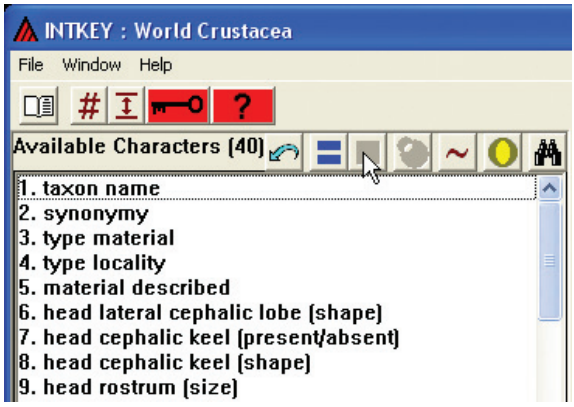


Figure 43. *Natural order* button selected.

3) Clicking on the *Differences between the specimen and the remaining taxa* button (Fig. 47) shows the differences between the specimen and the *Remaining Taxa* (the "specimen" is a name used for the character states that have been entered during the current session).

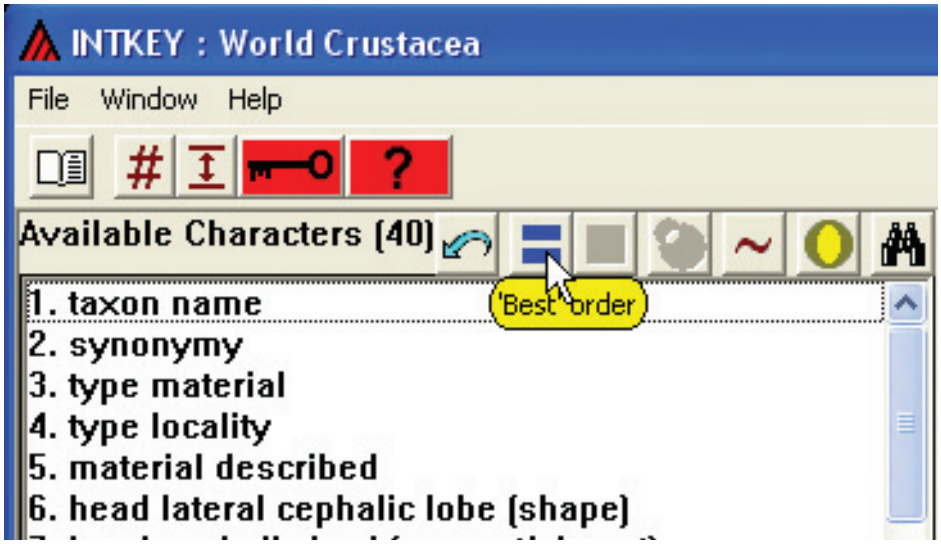


Figure 44. Selecting the *Best order* of characters.

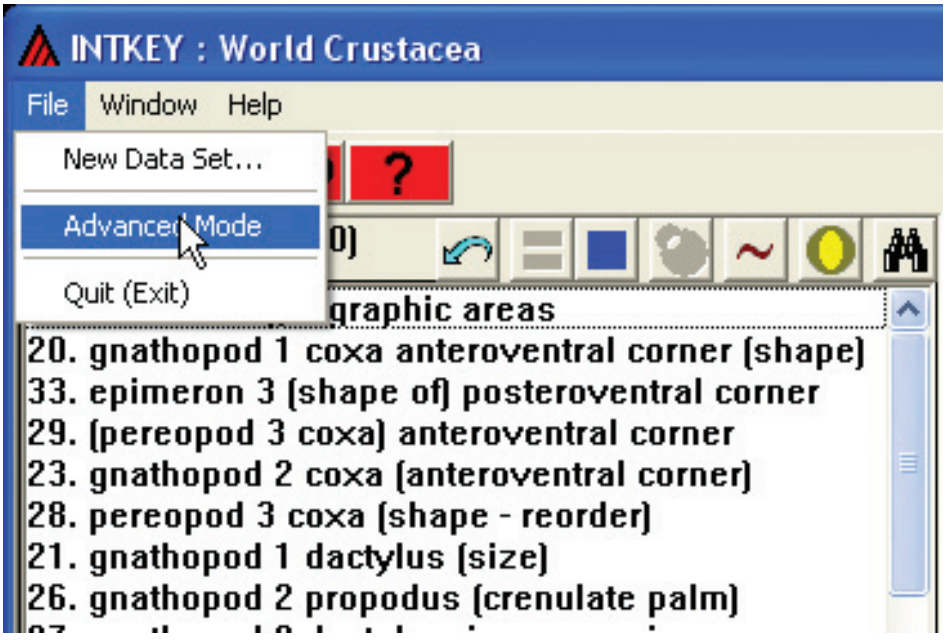


Figure 45. Switching into *Advanced mode*.

4) The *Set error tolerance* button (Fig. 48) allows for an increase in the number of mismatches during identification before taxa are eliminated. Allowing mismatches reduces the chances of the correct taxon from being eliminated because of a single misunderstood character state choice by the user or because the specimen has a slightly wider range of variation than has been allowed for in the data of its taxon. It makes a straight forward identification harder, but gives a more rigorous result.

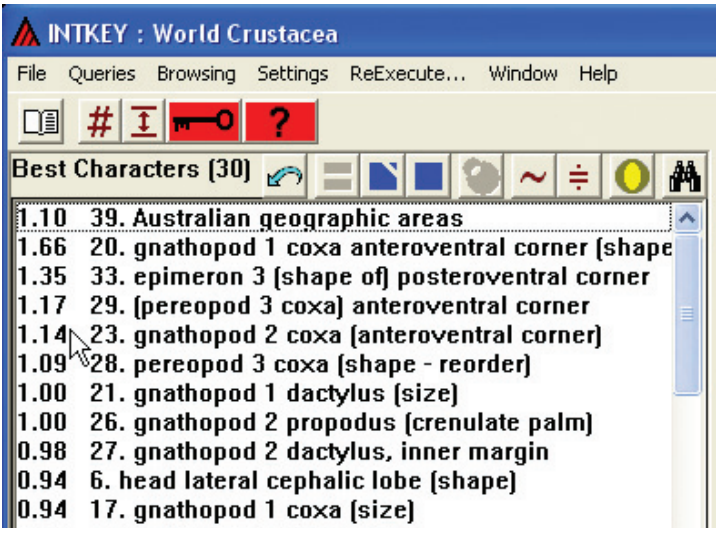


Figure 46. The separating power is shown.

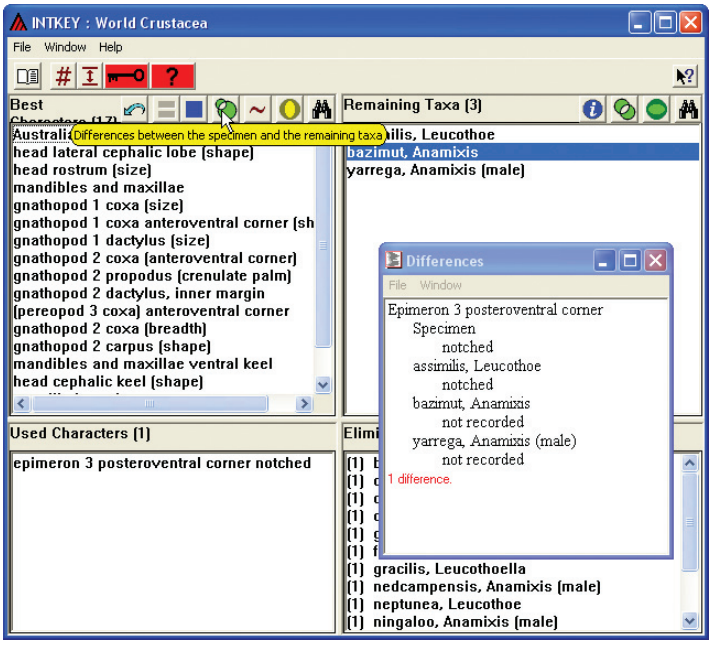


Figure 47. Differences between the specimen and the remaining taxa button.

5) *Use a subset of characters* button (Figs 49–51). These subsets are predefined in Intkey.ink with the DEFINE CHARACTERS command and limit the characters in groups for information retrieval. All is the default. For example the user may only want to use the mouthpart characters (Fig. 50) and clicks on *Head*. Then in the *Best Characters* window only these characters are available. Clicking *All* retrieves the full set of characters (Fig. 51).

6) The *Find text in characters* binocular button (Fig. 52) allows a text search in the characters. For the fullest search, tick the boxes for searching the states and used characters.

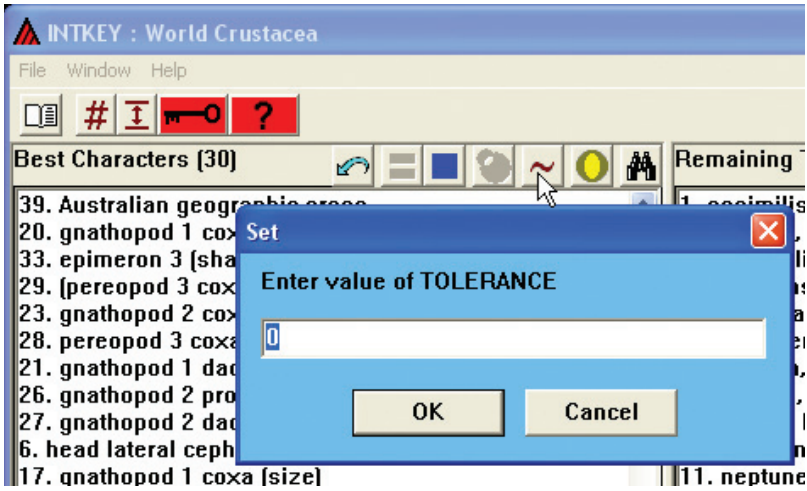


Figure 48. *Set error tolerance* button.

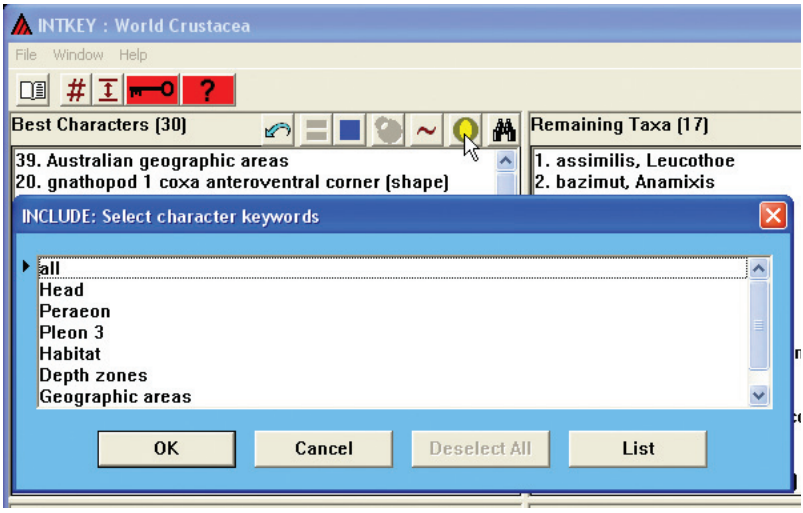


Figure 49. *Use a subset of characters* button.

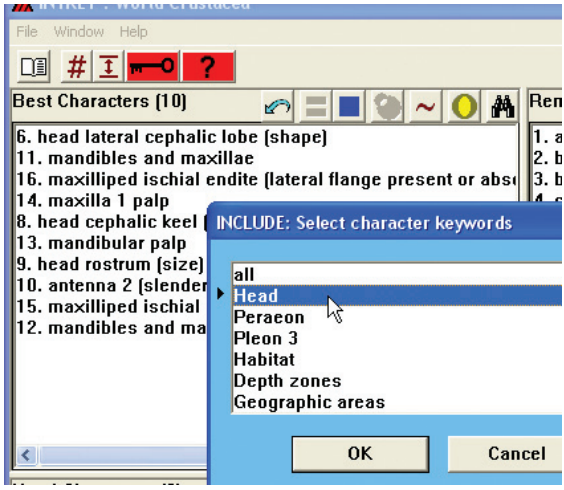


Figure 50. *Use a subset of characters* button.

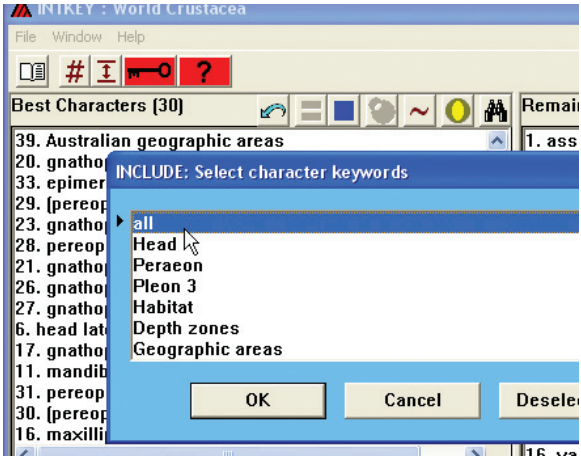


Figure 51. *Use a subset of characters* button.

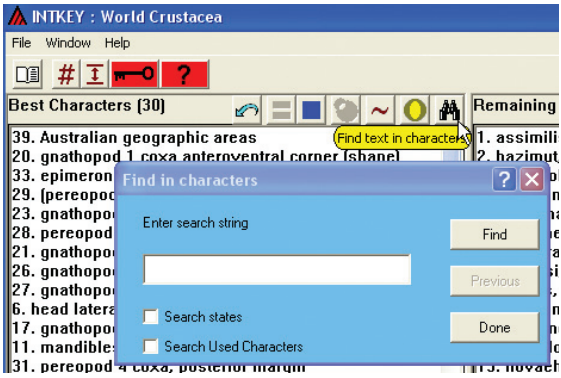


Figure 52. *Find text in characters* button.

The Remaining Taxa toolbar:

1) *Information about taxa* button: After selecting a taxon or a group of taxa a click on the blue Information button opens a submenu (Fig. 53). The selected taxon is shown in the upper text field. If several taxa are selected it is possible to browse through the taxa by arrow buttons or a direct selection can be made from the drop down list on the right side of the field.

In the left field of the dialog box a *Full description* of the selected taxon can be retrieved (Fig. 54). The full taxon descriptions only appear when they were previously created using the Action set *tonatsr* together with *ofiles* (see above). It is possible to diagnose the selected taxon by clicking on *Diagnostic Description* (Fig. 55). If the

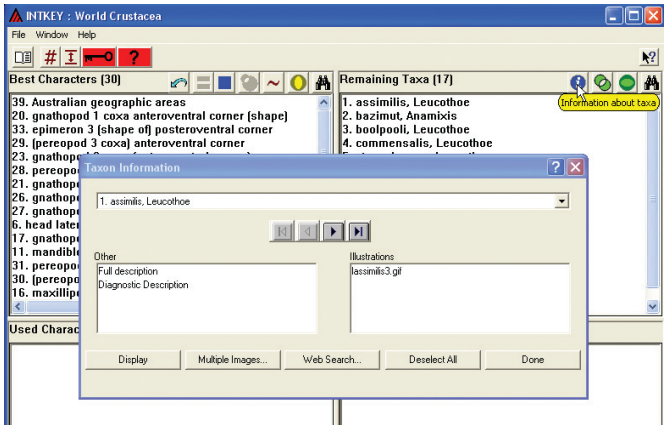


Figure 53. *Information about taxa* button.

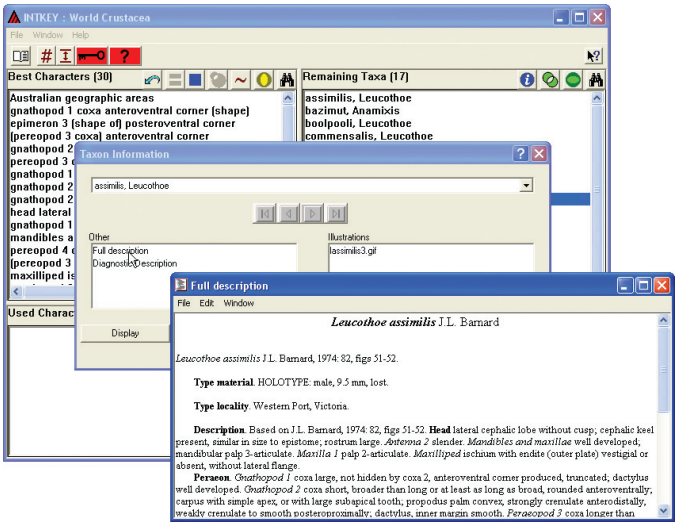


Figure 54. *Full description* of a taxon.

user wants to get a longer diagnosis it is possible to raise the diagnostic level (see user defined toolbar) (Fig. 56). In our example we raise the diagnostic level to *'3'*. Now Intkey produces more diagnostic characters. In Fig. 57 two species are diagnosed: for *Leucothoe boolpooli* Intkey produced three pairs of characters that are diagnostic for this species. For *Leucothoe assimilis* only diagnostic level 2 was reached.

By double clicking on the file names in the right field taxon illustrations can be opened (Fig. 58). Also several illustrations can be opened at the same time by marking them (clicking on the file names) and clicking on the *Display* button. If several taxa are selected a click on the *Multiple Images* button opens them all (use *Window/Close all* from drop down menu selection later in order to close them all).

A web search for the selected taxon can be invoked with the *Web Search* button.

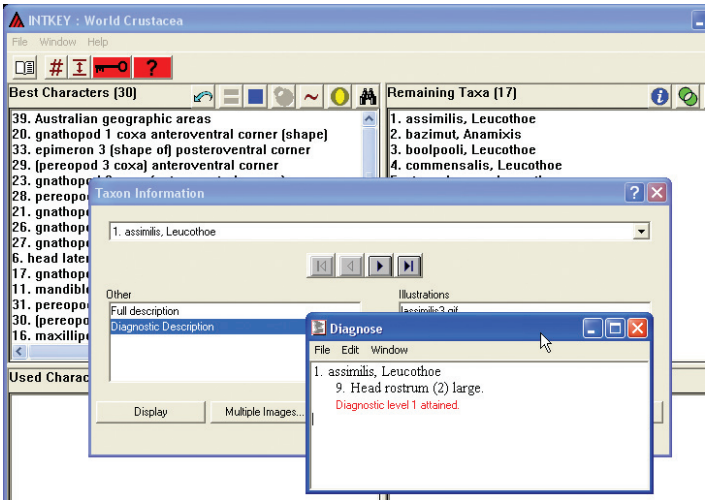


Figure 55. Diagnostic description.

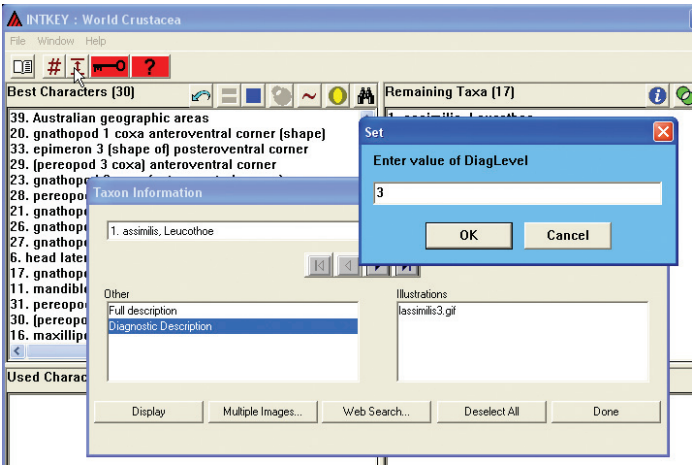


Figure 56. Raising the diagnostic level.

2) Clicking on the *Differences between taxa* button (Fig. 59) produces a window listing the comparison of several selected taxa (a non-contiguous set of taxa can be selected by holding down the control key while clicking the taxon names; contiguous sets can selected by using the shift key).

3) In Intkey.ink groups of taxa (e.g. genera) can be defined with the command DEFINE TAXA. These groups of taxa can be called up by selecting the allocated keyword. At a click on the *Use a subset of the taxa* button a dialog box appears where the selection of the keyword (= taxon or taxon group) can be made (Fig. 60).

4) The binocular button opens the *Find text in taxon names* dialog box (Fig. 61) which allows searches for text strings within the taxa.

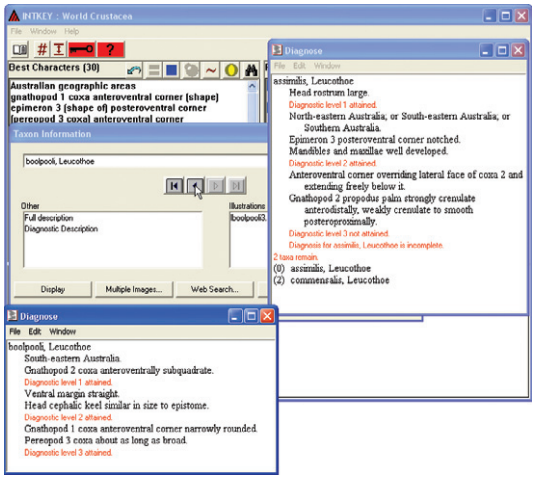


Figure 57. Diagnostic descriptions of two species where the diagnostic level has been raised to a value of '*3*'.

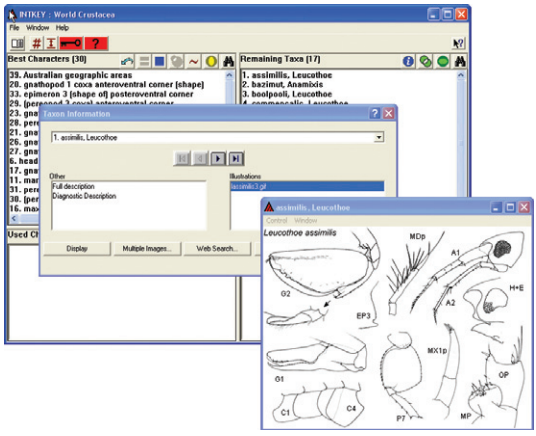


Figure 58. Display of a taxon image.

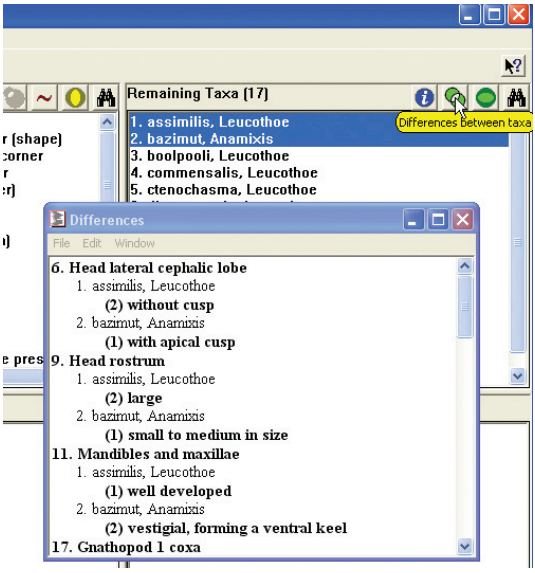


Figure 59. *Differences between taxa* button.

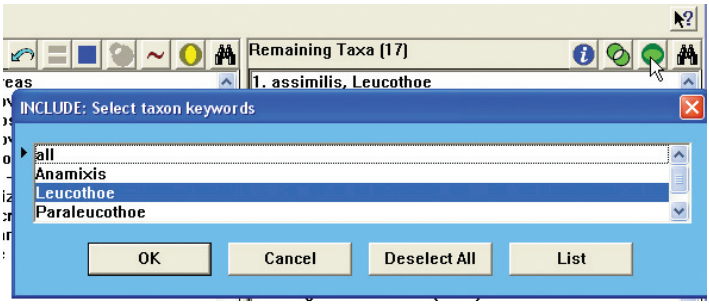


Figure 60. *Use a subset of the taxa* button.

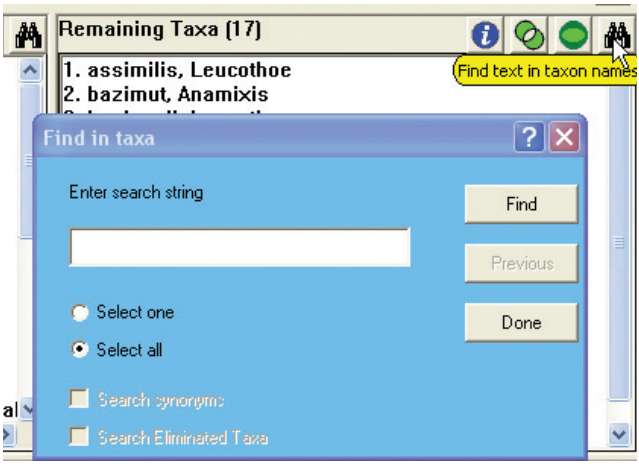


Figure 61. *Find text in taxon names* button.

Dichotomous keys

DELTA can produce keys as text outputs using the Confor directive file *tokey* (and files under the *Key* tab). This facility can produce dichotomous keys to large numbers of taxa and has some important advantages over traditional dichotomous key-making using manual methods. These include the ability to produce keys quickly in an automated process, the ability to consequently re-generate the key at any time, for example to update after making changes to taxa or characters in the database, the ability to make keys to subsets of the taxa, such as the species of a particular area, and using subsets of the characters for special purposes. The key program can also be set to use additional characters (confirmatory characters) at each step in the key if any are available.

One of the advantages of the DELTA dichotomous key-making program is that at each step characters are chosen according to a measure of best separation of the taxa, and this calculation is affected by factors that can be set, both overall or for each character. As the key generation occurs automatically when *tokey* is run, these settable factors provide only an indirect way to influence which characters are being used in the key and in which sequence they occur. DELTA prefers those characters (analagous to '*Best*' characters in Intkey, see below) that would split a set of taxa in nearly equally large groups. However, the user can raise the 'character reliabilities' in *tokey* and DELTA will use these characters for the construction of the key earlier.

This is a powerful and useful facility, details of which are given in the DELTA User Guide referred to earlier. In the following section we describe an alternative method for making small dichotomous keys using Intkey, which is less well documented elsewhere.

Making a dichotomous key using Intkey

In order to have full control on the characters being used for the construction of a key, we recommend using Intkey. This method works very well for making keys for a small number of taxa. Dichotomous keys are not necessary when it is possible to use interactive keys, but there are times when a dichotomous key may be required, for instance in a published paper.

The following instructions show two things at the same time: how to identify species using Intkey and how dichotomous keys are produced by a sequence of identification steps.

Using Intkey to help build such a key allows the author complete control of its construction. The process uses Intkey to keep track of characters and taxa instead of manual aids such as a table of characters, or the error-prone reliance on memory. It also takes advantage of the powerful Best function of Intkey to help in the choice of characters at each step, but aided by the users appreciation of the useability or ease of observing and understanding the differences between states of the characters. In what follows the user writes the key separately, although character text can be copied and pasted from a copy of the characters, which can be easily obtained from Intkey in Advanced mode from the menu: browse/characters/all.

Using the *AusLeucothoid* database as an example:

- Start up Intkey.
- Set to *Advanced mode* (drop down menu: *File/Advanced mode*).
- Set to interrogation mode, by clicking on the *red question mark button* (Fig. 63).

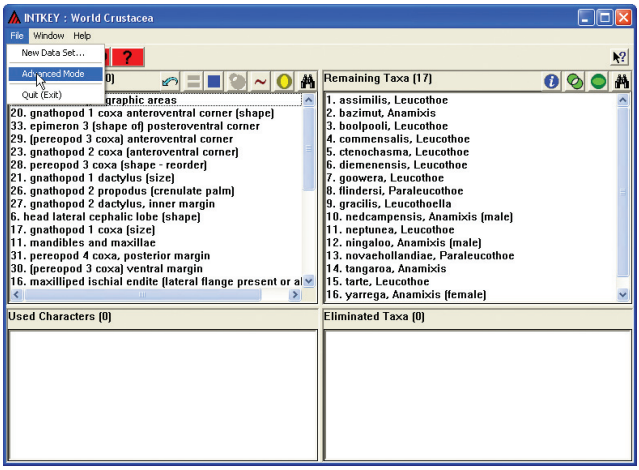


Figure 62. Switching into *Advanced mode*.

Figure 63. Interrogation mode.

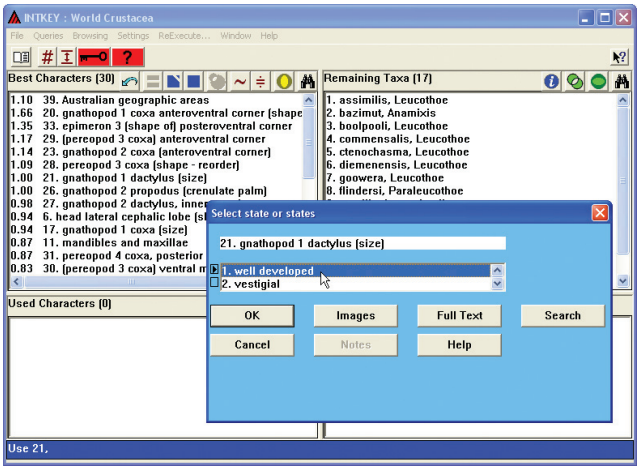


Figure 64. Character 21 is selected for the first step of identification.

- In the left hand column of the *Best Characters box* look for a *Best* score of around 1. This will divide the taxon set into halves. Look for a character that is clear and easy to recognise. For instance '*gnathopod 1 dactylus size*' (Fig. 64).
- Choose state 1 '*gnathopod 1 dactylus well developed*' and Intkey selects 8 species of *Leucothoe* and eliminates 9 other species, mainly *Anamixis* and *Paraleucothoe* (Fig. 65).
- Continue with the *Leucothoe* selection and ignore the eliminated taxa for now.
- So the first dichotomy in your key is:

1. Gnathopod 1 dactylus well developed **2**
– Gnathopod 1 dactylus vestigial ..... x →

- Look again for a *Best* score of around 1, again emphasizing clarity and easy recognition. Select for instance, '*pereopod 4 coxa posterior margin with a posteroventral lobe*' (Fig. 66). Intkey selects 4 species.
- So the second dichotomy becomes:

2. Pereopod 4 with well developed posteroventral lobe 3
– Pereopod 4 without posteroventral lobe..... x → 5

- At this stage a *Best* score of 1 is no longer available (Fig. 67). Epimeron 3 with a score of 1.5 is not optimal because it is a 4 state character. But the epimeron is easy to see and only three states are scored for the remaining taxa (which can be deter-

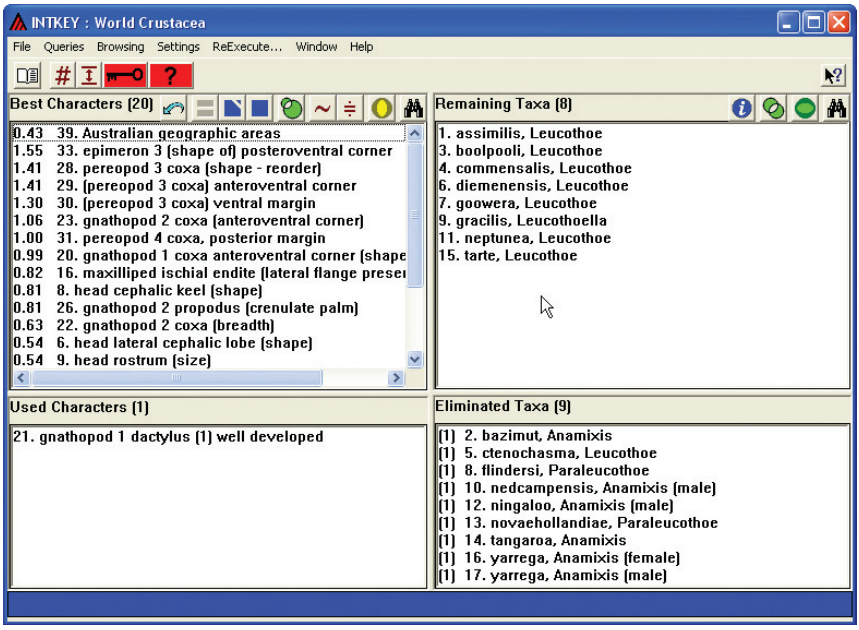


Figure 65. The first character separated the taxa into two groups of about equal size. For the *Eliminated Taxa* the selected character state did not match.

mined by examining the data for that character for the remaining taxa by using the Intkey menu to click Queries/Describe/Remaining (taxa)/Available (characters)/List, and selecting the relevant character – the results appear in a separate window). So a trichotomy is possible. We now reach several endpoints in the key which we obtain by following each state of the character then going back (see below for how to do this). It might be possible to represent the generic name *Leucothoe* by *L.*, but in this key there are two genera, *Leucothoe* and *Leucothoella* starting with the same letter so it is better to use the entire species name.

3. Epimeron 3 posteroventral corner notched .....*Leucothoe assimilis*
– Epimeron 3 posteroventral corner narrowly rounded....*Leucothoe goowera*
– Epimeron 3 posteroventral corner subquadrate 4

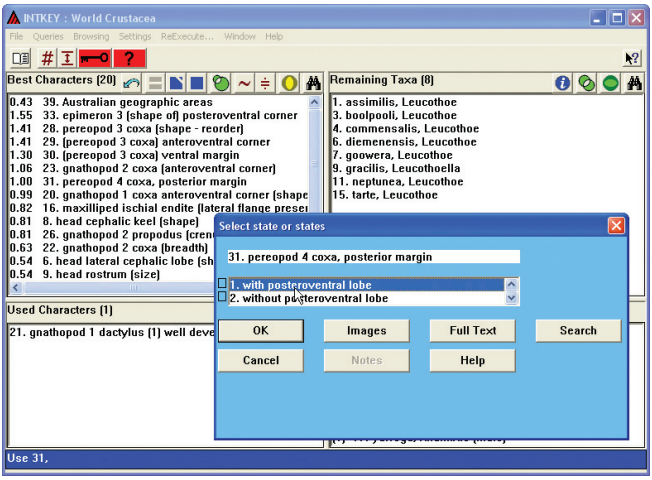


Figure 66. Character 31 is another well visible trait, it has the ideal *Best* index of 1.0.

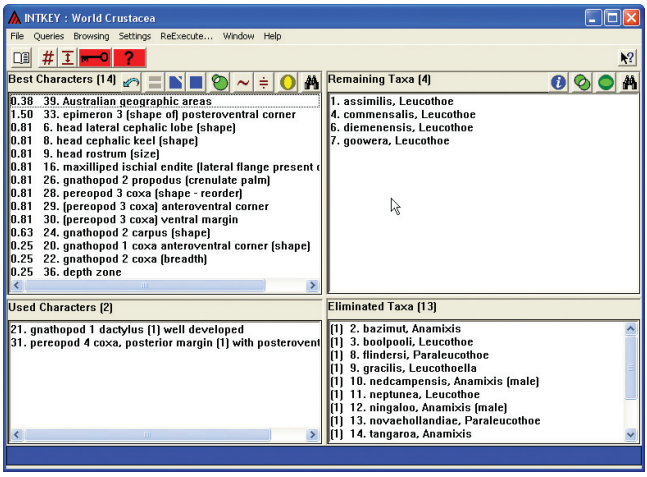


Figure 67. Character 31 separated the 8 *Remaining Taxa* into halves by eliminating 4 taxa.

- There are now four characters which will separate the remaining taxa. In this case it might be possible to select two of the *Best* list: '*gnathopod 2 carpus shape*' and '*pereopod 3 coxa ventral margin*' (Figs 68–70).

4. **Gnathopod 2 carpus with simple apex. Pereopod 3 coxa ventral margin slightly convex ..... *Leucothoe commensalis***

– **Gnathopod 2 carpus with large subapical tooth. Pereopod 3 coxa ventral margin oblique and slightly concave ..... *Leucothoe diemenensis***

- It is now necessary to go back to couplet 2 and pick up the four taxa with a '*well developed gnathopod 1 dactylus*' and '*without a posteroventral lobe on the coxa of pereopod 4*'. This can be done by clicking on the epimeron 3 character in the bottom left "used char-

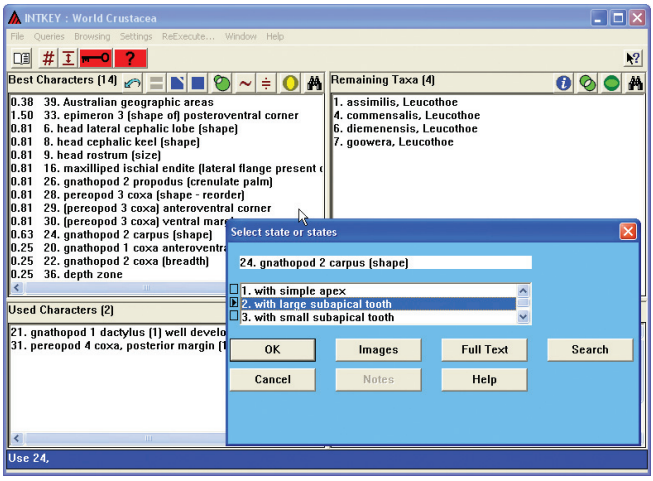


Figure 68. Character 24 again splits the *Remaining Taxa* into two equally large groups.

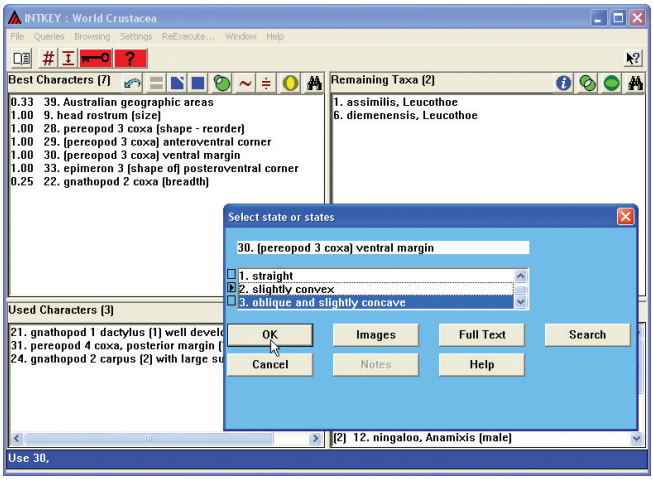


Figure 69. Using a fourth character.

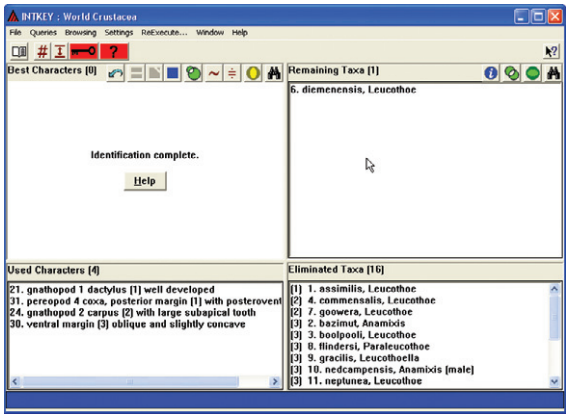


Figure 70. The identification sequence for one part of the key is complete. Now we go back a few steps to continue to work on the key.

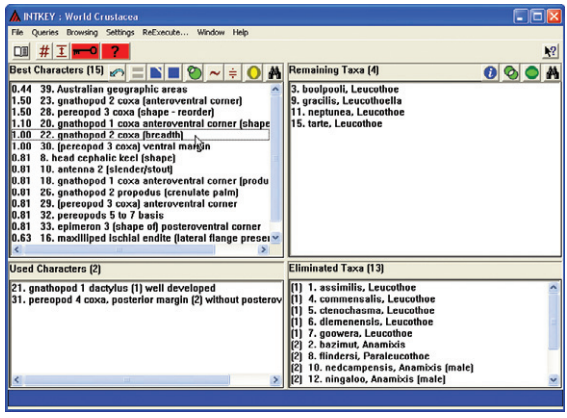


Figure 71. The second group of taxa is used to create the dichotomous key.

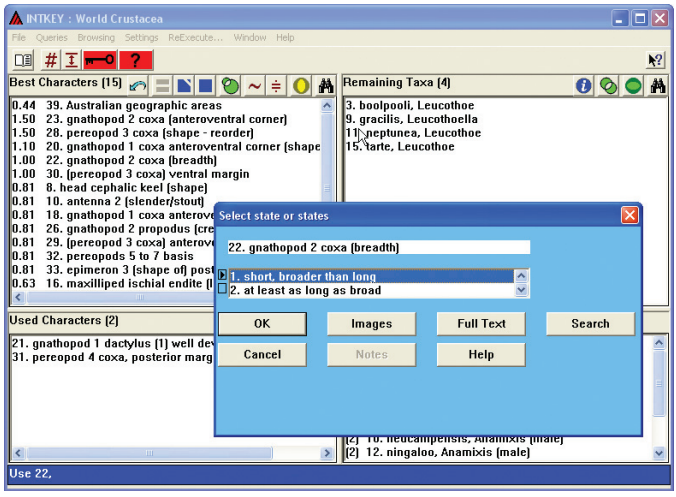


Figure 72. Character 22 has a *Best* index of 1.0 which is ideal for identification.

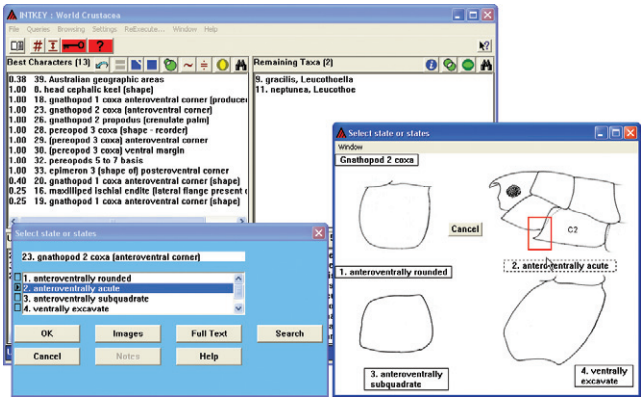


Figure 73. Clicking on the *Images* button during the selection of character states will show the illustrations for all character states.

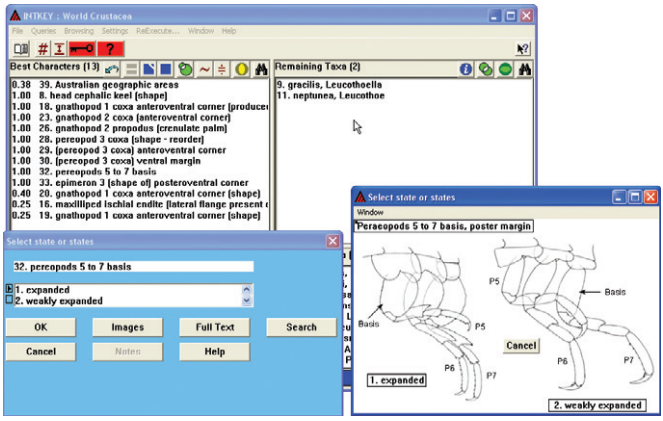


Figure 74. Clicking on the *Images* button during the selection of character states will show the illustrations for all character states.

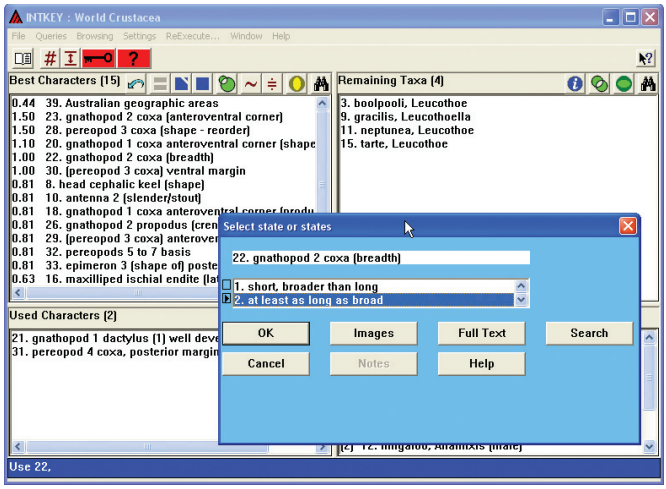


Figure 75. Selecting character states.

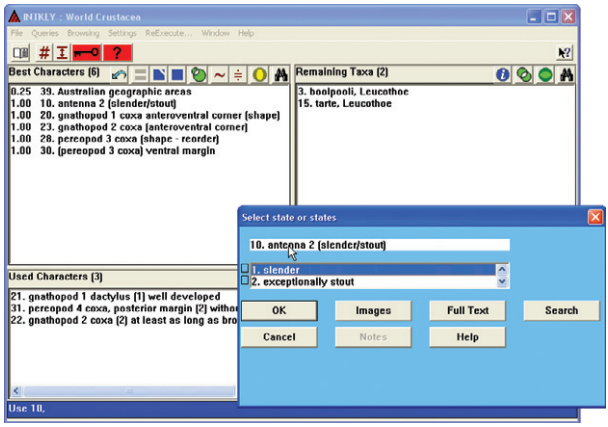


Figure 76. Selecting character states.

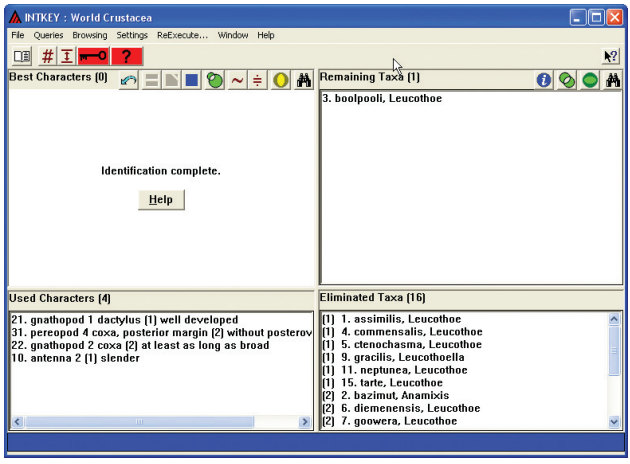


Figure 77. For the second set of taxa the identification is finished and another endpoint of the key is reached.

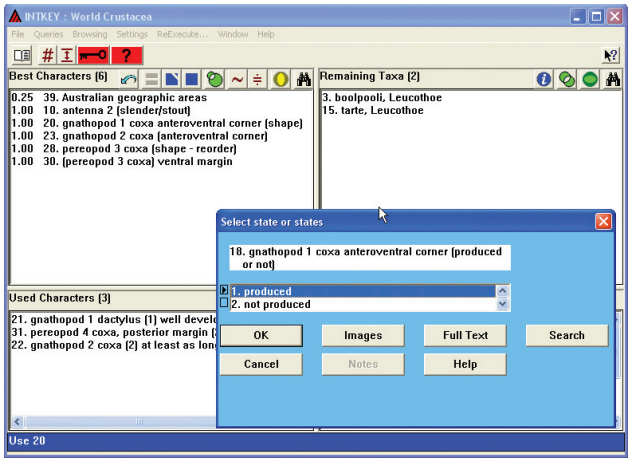


Figure 78. Character 20 is connected to character 18. Clicking on character 20 will open character 18 first.

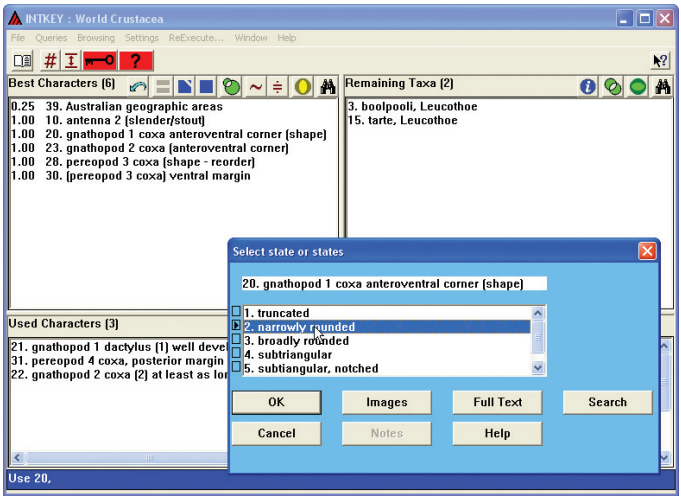


Figure 79. Character 20 is dependent on character 18. After selecting the character state '*produced*' in character 18, the character state dialog box appears for a selection of a character state of character 20.

acter" box. Then click on the highlighted state in the upper left "select state or states" box to deactivate it and select OK. Repeat for character 30, pereopod 3 coxa ventral margin. Then click on the pereopod 4 character in the "used character" box, deactivate the highlighted state by clicking on it and select *without posteroventral lobe* (Fig. 71).

- There are two *Best* characters with a value of 1, either or both of which will separate the remaining taxa: *gnathopod 2 coxa breadth* and/or *pereopod 3 coxa ventral margin*. Unfortunately the characters do not give the same set of taxa. In this case it is better to choose the most conspicuous character, probably '*gnathopod 2 coxa breadth*'. So the second part of couplet 2 is labelled as couplet 5 which becomes.

5. Gnathopod 2 coxa short, broader than long..... 6
– Gnathopod 2 coxa at least as long as broad 7

- There are 9 choices to separate the remaining taxa. By comparing the whole animal illustrations (using the blue information button) the best choices might be '*gnathopod 2 coxa anteroventral corner*' and '*pereopod 5 to 7 basis expanded*' (Figs 72–74).

6. Gnathopod 2 coxa anteroventrally acute. Pereopods 5 to 7 basis weakly expanded.....*Leucothoella gracilis*
– Gnathopod 2 coxa anteroventrally rounded. Pereopods 5 to 7 basis expanded.....*Leucothoe neptuna*

- Now go back to couplet 5 by clicking on '*gnathopod 2 coxa*' in the '*Used characters*' box and select the second choice '*at least as long as broad*' (Fig. 75). The remaining two taxa will separate on five choices. In this case there are two conspicuous

characters that could be used to separate these taxa: the robustness of antenna 2 (Figs 76, 77) and the shape of the anteroventral corner of coxa 1 (Figs 78, 79). The latter shows how Intkey displays a controlling character (18) which logically needs to be answered before its dependent character (20).

7. Antenna 2 slender. Gnathopod 1 coxa anteroventral corner produced and narrowly rounded ..... *Leucothoe boolpooli*
– Antenna 2 robust. Gnathopod 1 coxa anteroventral corner produced and truncated..... *Leucothoe tarte*

- The first half of the key is now complete. The developer must now go back to couplet one and start construction of the second part in exactly the same way. This can be done by restarting the interrogation using the arrow button on the *Best Character* menu bar, then selecting: '*gnathopod 1 dactylus vestigial*' and beginning at couplet 8.

Discussion

DELTA has the potential to revolutionize taxonomy. In individual databases, institutional databases and global databases the knowledge of plant and animal systematics can be stored, maintained as long-term databases and transferred to users of taxonomy (e.g. ecologists). Better than printed media, which are already out of date when they reach the scientific community, DELTA databases allow the addition of new taxa over the time and allow the continuous growth of knowledge over generations of scientists. Also illustrations and multimedia data, such as sound files can be included in the database.

DELTA is an extremely powerful though very complicated piece of software. This is partly due to its developmental history, which dates back into the dark ages of card decks and mainframe computers. Although editing of the main data is now handled via a modern interface (the DELTA Editor), processing of the data (i.e. output) is still handled by the original programs (now modified to run in Windows), which may be run from within the DELTA Editor. This make the processing of the data rather cumbersome, particularly as some kinds of error are only detected at the 'processing' stage.

DELTA is also complicated due to the complexity of the DELTA format itself (see Dallwitz and Paine 1993) and the diversity of output functions. More than 80 commands can be controlled by the user in a "command line" style. This has to be done without the help of menus. It is very flexible, but difficult to memorize.

In a future version, perhaps some of these problems will be overcome by introducing a menu driven interface for the output functions. There are discussions for a future upgrade of DELTA in connection with the project Atlas of Living Australia. In the meantime we tried to facilitate the access to DELTA by introducing a new sample database "Deltablank.dlt". In this we only included the essential Action sets. Additionally we commented all commands in the Action sets of this database.

Two of us (J.L. and T.M) were teaching DELTA workshops for systematists, each more than a week long, all over the world in order to promote the idea of DELTA. This was very successful, however, over time many of the participants forgot the details, when not using DELTA continuously.

So the idea came up to write this introduction for the beginner and as a refresher course for the advanced user.

The great thing about DELTA is its multi-functionality. It can be used as a database for information retrieval and as a basis for generating printed publications, keys and matrices for phylogenetic studies. DELTA develops its full potential by the possibility to publish contents, such as interactive identification files (Intkey), over the internet. One very successful example is crustacea.net (http://www.crustacea.net).

These Intkey files can be published also on CDs as valuable additions for printed media (examples are Beetle Larvae of the World (Lawrence et al. 1993); The Families of Flowering Plants (Watson and Dallwitz 1993); Census of Antarctic Marine Life (De Broyer et al., 2007); Amphipoda of the Great Barrier Reef (Eds Lowry & Myers, 2009).

Acknowledgements

Thanks to a grant of the German Research Foundation (DFG) O.C. could travel to Australia for the joint effort to write this manual (project CO 165/7-1). The authors thank the two reviewers who improved the manuscript by their valuable comments. Appreciation is also extended to Dr. Mike Dallwitz who improved the manuscript considerably by his comments.

References

Dallwitz MJ (1974) A flexible program for generating identification keys. Syst. Zool. 23: 50–57.

Dallwitz MJ (1980) A general system for coding taxonomic descriptions. Taxon 29: 41–6. Also available at http://delta-intkey.com/www/dallwitz-1980.htm.

Dallwitz MJ (2000 onwards) An Intkey example: identification. http://delta-intkey.com/www/id.htm.

Dallwitz MJ (2009) Overview of the DELTA System. http://delta-intkey.com/www/overview.htm.

Dallwitz MJ, Paine TA (1993 onwards) Definition of the DELTA format. http://delta-intkey.com/www/standard.htm.

Dallwitz MJ, Paine TA, Zurcher EJ (1993 onwards) User's guide to the DELTA System: a general system for processing taxonomic descriptions. 4th edition. http://delta-intkey.com/www/uguide.htm.

Dallwitz MJ, Paine TA, Zurcher EJ (1999 onwards) User's guide to the DELTA Editor. http://delta-intkey.com/www/delta-ed.htm.

De Broyer C, Lowry JK, Jazdzewski K, Robert H (2007) *Catalogue of the Gammaridean and Corophiidean Amphipoda (Crustacea) of the Southern Ocean with distribution and ecological data*. Bulletin de l´Institut Royal des Sciences Naturelles de Belgique, Brussels.

Lawrence JF, Hastings AM, Dallwitz MJ, Paine TA (1993) Beetle larvae of the world. CD-ROM, Version 1.0 for MS-DOS. CSIRO Publications, Melbourne.

Lowry JK, Myers AA (2009) Benthic Amphipoda (Crustacea: Peracarida) of the Great Barrier Reef, Australia. *Zootaxa* 2260:1–930.

Watson L, Dallwitz MJ (1993) The families of flowering plants. CD-ROM, Version 1.0 for MS-DOS. CSIRO Publications, Melbourne.

Appendix

The following two instructions (Scanning, Format conventions) have been written by Roger Springthorpe (Australian Museum, Sydney).

1) Scanning black and white images using Adobe Photoshop

These instructions are for **Adobe Photoshop 5.0** and **HP Scanjet 6200C** scanner for Windows NT.

Turn on scanner, then computer. (If computer is on and scanner is off, shut down computer, switch off power, switch on scanner, then computer. Scanner won't work unless this sequence is followed).

Select **Settings, Control Panel, Display, Settings** and check that **Colour Palette** is set to 65536 colours, **Font size**: Small fonts and **Desktop Area**: 1024x768 pixels.

Start up Adobe Photoshop.

Scansning an image

Images need to be A4 size. A clean photocopy is adequate but scanning of originals is best.

Place image on scanner, face down.

From pull down menu, choose **File, Import,** select **TWAIN_32.** (This activates the scanning software).

The HP Precision Scan Pro window appears and a preliminary scan is carried out. Next set up scanning criteria: select **Output Type, Greyscale, OK**. Then select **Tools**, **Output Resolution** set to 300dpi and click on x to close. Then select **Scan** from main menu then **Save Settings** and type in a name (eg "DELTA") for these settings then **OK**. This saves the settings for subsequent use. Choose **Scan,** select **Load settings.**

Select **"DELTA"** from pull down menu, select **Load**. A cross appears in the viewing window. Click and drag to outline the area to be scanned.

Choose **Scan, Place Image**.

Image appears in Photoshop as "Untitled-1" @ xx%.

Load Settings must be used for each scan or else the default settings will be used.

Saving files. From pull down menu choose **File, Save as,** select Folder for the file to be saved in, insert unique **<filename>.PSD**. Select **OK.**

The image can now be manipulated in Adobe Photoshop as a *.PSD file.

For Subsequent saves use **File, Save.**

To incorporate files in the DELTA program *.GIF files are required.

After the image has been manipulated (see following paragraphs) save the *.PSD file but do not close it. From pull down menu choose **Image, Mode,** select **Indexed colour**. A dialogue box appears: "Flatten Layers?" - select **OK.**

From pull down menu choose **File, Save as,** select folder (preferably different from the *.PSD files) and save as *.GIF. Then close *.PSD file without saving. This will preserve the psd file and layers etc for future editing if required. GIF files cannot be manipulated like PSD files so if changes need to be made use the original PSD file and then make a new GIF file.

Manipulating an image

Creating a new file. Select **File, New.** Enter new file name and pixel dimensions, then OK.

This creates a new blank Photoshop image into which the required parts are copied and pasted from the original scans.

Tools. These will be needed to manipulate the image

From pull down menu choose: **Window** and select **Show Tools, Show Brushes, Show Layers, Show Options** and **Show Status Bar.**

Image and canvas size. The image can be reduced in size to fit on the screen. It is desirable to retain the original scanned image as it may be required for other purposes later. Once an image is reduced it can be enlarged but the quality suffers. Start from large and work down.

Canvas size can be enlarged or reduced without affecting the image.

To change image size select **Image, Image size**. The **Image Size** dialogue box appears. Select **Constrain proportions,** type in the **pixel dimensions** for height or width - the other dimension is automatically entered. Select **OK.** It should be noted that image scaling may adversely affect the image quality. The best results are obtained if integer increments are used e.g. 25%, 50% etc rather than 33% or 29% etc.

To change canvas size select **Image, Canvas Size.** The Canvas Size dialogue box appears, enter both width and height dimensions required, select **OK.** If the canvas size is being reduced a prompt will appear select **Proceed.** If the image is chopped off use the undo tool and repeat the process.

Alternatively, the **Crop tool** lets you crop an image by dragging over the area you want to keep.

Magnification. The scanned image is usually too large to fit on the screen at 100%.

% image size is shown on the status bar at bottom left corner of the screen and may be changed by selecting this box and entering a suitable value. Alternatively, from the **Tool bar** select **Zoom Tool** and then select the image to increase the magnification in set increments.

Adjust levels. Use this to make gradual adjustments to the brightness, contrast, and midtones in an image. Most of the grey in a background can be whitened and/or the image darkened using the levels adjustment.

From pull down menu choose: **Image, Adjust, Levels.** The **Levels** window appears. Adjust **input** levels. This requires some experimentation to get a satisfactory image. Try something like 45 - 1.00 - 240. When finished select **OK.** Alternatively, select the **Auto Levels** option - it's quicker but you have less control over the output.

Layers. If a scanned page consists of multiple parts and needs to be recomposed ie parts repositioned or removed etc, it is necessary to copy and paste each part onto its own layer. Text also requires a separate layer or layers. It is necessary to have the **Show layers** dialogue box visible to work with layers and the required layer must be highlighted in this dialogue box.

From the pull down menu choose **Layer, New, Layer.** The **New Layer** dialogue box appears, enter a name for the layer. This is useful to identify a particular layer to manipulate if there are multiple layers.

To copy and paste part of an image onto a new layer select **Lasso Tool** from the tool bar**.** Outline the required part of the image. In **Layers** dialogue box, select required layer. From pull down menu choose **Edit, Copy**. In **Layers** dialogue box choose the **new layer**. From pull down menu choose **Edit**, **Paste**. Image is now on a new layer.

When copying from one file into another the new layer is automatically created. Double click on the layer in the Layers dialogue box and type in a name for the layer.

Move. To reposition a layer (or image) select the **Move Tool** from the tool bar and click and drag the image.

Transform. A layer (or image) can be rotated, scaled, flipped, skewed, or distorted.

Select the required layer in the **Layers** dialogue box.

Select **Layer, Transform,** then select option required.

Alternatively select **Layer, Free Transform.** This allows any combination of these transformations in one operation.

Eraser. Use the eraser to clean up an image.

Select the required layer in the **Layers** dialogue box.

Select **Eraser Tool** from tool bar, select size of eraser from **Show Brushes** dialogue box.

To clear the background, choose **background layer** in **Layers** dialogue box, then double click the **Eraser Tool** in the tool bar and choose **Erase layer** from the **Eraser Options** dialogue box.

Text. Text is added to the image on a new layer. It can therefore be moved, transformed, erased, etc using the various tools.

Select the **Type Tool** from the tool bar. Click on the image window and the **Type Tool** dialogue box appears. Select various options for font, size etc. Write text, click **OK**.

To add male/female symbols, change font to male and female symbols. Type **f** for female and **m** for male.

Arrows. Arrows can be inserted using the Line Tool.

Select Line Tool from tool bar. Select various options from **Line Tool Options** dialogue box eg arrow head shape, line thickness etc.

Undo. This option is very useful to undo the previous action. From the pull down menu select **Edit, Undo**.

2) Format conventions for illustrations

Text

Text is required for both taxon and character images and inserted either in Photoshop or in the DELTA Editor in the image dialog of the taxon editor and the character editor.

Taxon images are prepared in Photoshop and text is inserted at this stage.

Taxon names: top left or centre aligned. Font: Arial or Helvetica, regular (higher taxa names) or italics (species names), 14 pixels, anti-aliasing set to smooth.

Labels on taxon images: Font: Arial or Helvetica, regular, 12 pixels, anti-aliasing set to none.

Acknowledgement: Font: Arial or Helvetica, regular, 10 pixels, anti-aliasing set to none.

Page number (ie 1 of 2, 2 of 2 etc) inserted using Intimate, bottom right aligned.

Arrows for indicating blow-ups: weight 2 pxl, mode normal, opacity 100%; shape: width 400%, length 800%, concavity 20% (see fig. 1)

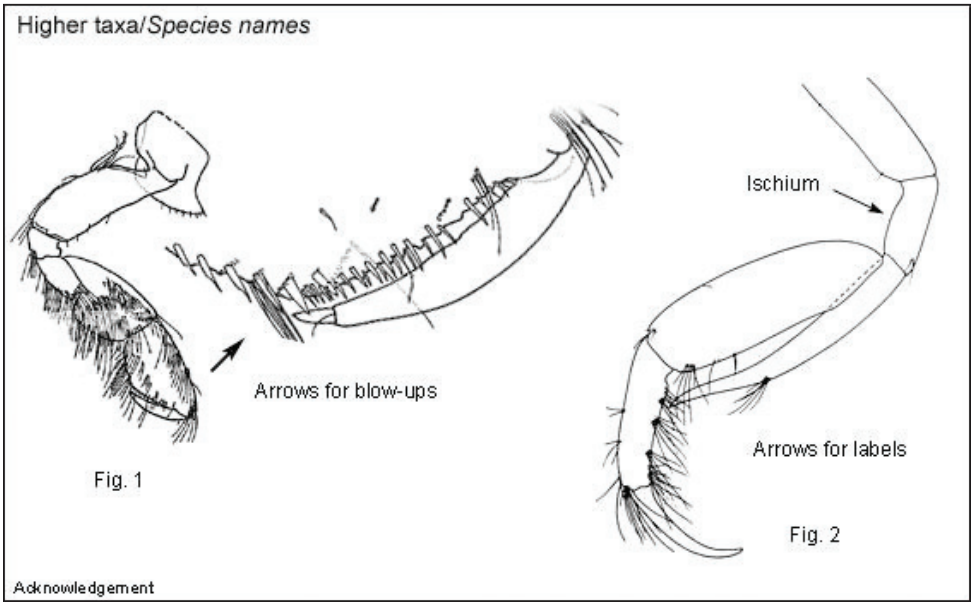


Figure 80. Example for character images.

Arrows for indicating a part being labelled: weight 1 pxl, mode normal, opacity 100%; shape: width 600%, length 1000%, concavity 20% (see fig. 2).

Male, female symbols: regular, 18 pixels, anti-aliasing set to smooth.

Character images are prepared in Photoshop and text for character title and states is inserted using the DELTA Character editor.

1. Labels on character images: Font: Arial or Helvetica, regular, 12 pixels, anti-aliasing set to none.
2. Arrows for indicating blow-ups: weight 2 pxl, mode normal, opacity 100%; shape: width 400%, length 800%, concavity 20% (see fig. 1).
3. Arrows for indicating a part being labelled: weight 1 pxl, mode normal, opacity 100%; shape: width 600%, length 1000%, concavity 20%.
4. Male, female symbols: regular, 18 pixels, anti-aliasing set to smooth.