Add semantic categories to your dictionary
Bruce Waters   PNG
June 1992

MacLex, version 2.1, now supports the Lowe and Nida semantic categories. Or you can use your own categories. The programme handles up to three levels - that is, categories, subcategories, and their subcategories.

MacLex is now distributed with two files of semantic category data. The first is an exact reproduction of Lowe and Nida's two levels of categories - over six hundred of them. The other file has the same data, but regrouped and with eight categories added at the top level to make it a three-level table. This file is a little easier to use because it puts up shorter lists of catergories to choose from.

So not only can you now use Maclex to create and manage dictionary records, sort them and reverse them, Maclex now gives you powerful features for adding semantic categories to your dictionary entries.

To insert categories into existing records, the user chooses a **Define Categories...** command. This puts up a window with a list of the first level of categories. Double-click on your choice and a second window with its subcategories will appear. Likewise a third. The screen picture illustrates what the screen looks like when the third window has been put up and a

category clicked. Each window obscures the preceding window except for a narrow strip at the left.

When your last choice is made, the windows disappear, the cursor shape changes, and the user can then open any record for editing. An **Insert Categories** command then writes the categories to the record - wherever the cursor or selection range is, using appropriate standard format markers for each level's category.

Notice the windows with the category lists are offset so that lower ones can be clicked if the user changes his mind about previous category or subcategory choices. Window widths vary according to the width of the categories they contain.

There are commands to load a semantic category table, enable earlier choices, disable the current choice, and to convert the form of the category data in records from one of three possible formats to any of the others. What the latter means, briefly, is as follows. Notice the screen picture shows that each line has an A, B, C, ... etc "prefix" - this is how Lowe and Nida's categories appear. Into records you can have MacLex insert just the prefixes, or just the category names (which follow the prefixes), or the full prefix plus category name.
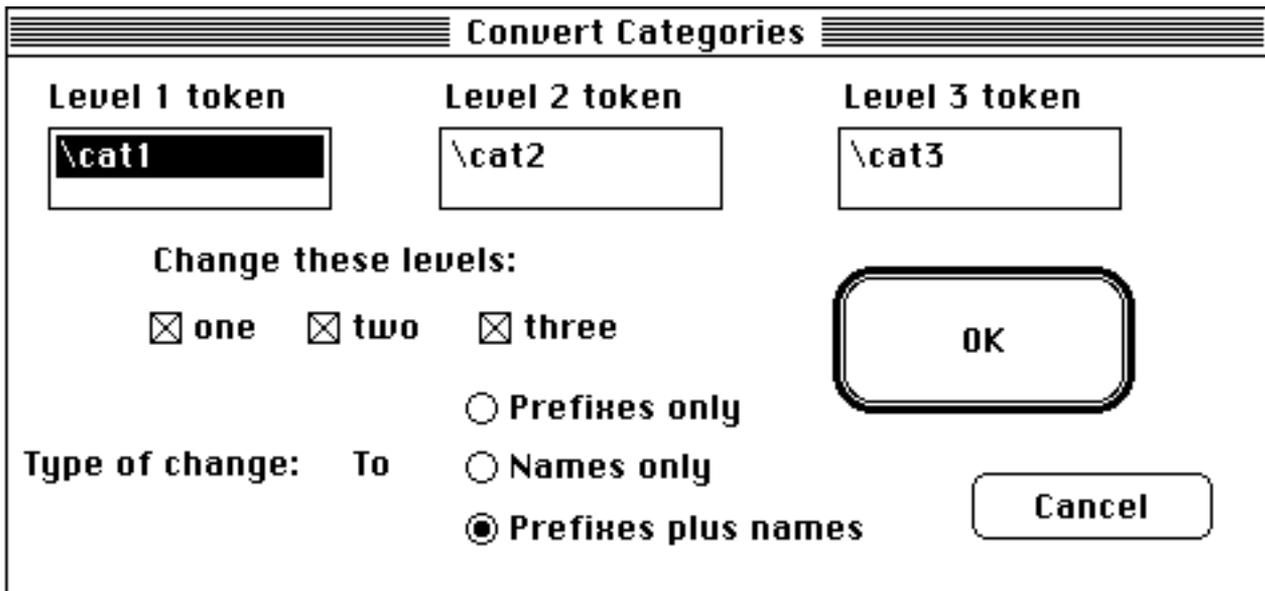
The dialog controlling the standard format markers to be used, the number of hierarchical levels to use (and hence how many hierarchical windows are shown for a **Define Categories...** command), and what kind of data you want to be written to fields in records, is shown below.

```
╔════════════ Semantic Category Options ════════════╗
║                                                     ║
║                  Level 1      Level 2      Level 3  ║
║                 ┌────────┐   ┌────────┐   ┌────────┐║
║ Semantic Category Tokens: │\cat1 │  │\cat2 │  │\cat3 ││
║                 └────────┘   └────────┘   └────────┘║
║                                                     ║
║ Number of levels to use:      ○ 1      ○ 2     ● 3  ║
║                                          ┌─────────┐║
║              ○ fields with Prefixes only │   OK    │║
║ Category type ● fields with Names only    └─────────┘║
║              ○ fields with Prefixes plus Names ┌──────┐║
║                                          │ Cancel │  ║
║                                          └──────┘    ║
╚═════════════════════════════════════════════════════╝
```

The other dialog which MacLex uses for controlling semantic category functions is shown below. It controls the conversion process. The "convert" operation allows you to change your mind at any time about what is to appear in the semantic category fields in records, and have MacLex do the necessary conversions for you automatically. You would use it, for example, to instruct MacLex that you wanted semantic category fields in records to have, say, just the "prefix" information.

When you dispose of the dialog, a **Convert Categories** command would do the actual searching for a bundle of categories to convert, and each time put up a dialog with buttons to **Convert**, **Skip**, or **Cancel** - the fields found appear highlighted for your convenience while this latter dialog is up. Alternatively an **Auto Convert Categories** command allows you to do the conversions without being asked for your response each time.

```
╔══════════════ Convert Categories ══════════════╗

  Level 1 token        Level 2 token        Level 3 token
 ┌──────────────┐     ┌──────────────┐     ┌──────────────┐
 │ \cat1        │     │ \cat2        │     │ \cat3        │
 └──────────────┘     └──────────────┘     └──────────────┘

   Change these levels:                    ╭───────────────╮
                                           │               │
   ⊠ one    ⊠ two    ⊠ three              │      OK       │
                                           │               │
                  ○ Prefixes only          ╰───────────────╯

 Type of change:   To  ○ Names only        ┌───────────────┐
                       ● Prefixes plus names│    Cancel     │
                                           └───────────────┘

╚════════════════════════════════════════════════╝
```

MacLex also allows you to search for a category together with the subcategories appearing with it. A **Find Categories** command does this operation, and a **Find Categories Again** command allows you to search repeatedly for the same bundle of category fields.

Other features added to MacLex since version 1.1d are as follows. Sorting now does not require the user to first create a destination folder. MacLex now does all needed file and folder manipulations, and the sorted files end up in the same folder as the input files, while the input files get automatically deleted. This operation can also be safely aborted, and MacLex will restore all the original files intact. MacLex also can now safely abort the following lengthy operations when the user types "command period:" sorting, exporting, reversing, replacing all, and automatic conversion. Known bugs have also been fixed.

So, now you can combine Maclex's powerful semantic category functions with its reversal functions and produce very useful reversed dictionaries. It's never been so easy!