

Sort Lists to Replicate the Function of an Index
By Anders R. Sterner
TPF&S Consulting

Some lists are better ordered alphabetically than by order of appearance. You can get that using an index, but you can have only one index in a document (or, indeed, in any collection of master and subdocuments). You can use a list to accomplish much the same thing, by generating once (to get the [End Def] code, and then putting a unique non-printable ASCII code (such as ASCII 182) immediately before the List Definition and immediately after the [End Def] code for that list.

You then run the included LSTSRT.WPM macro to cause the sort. That macro takes into account the following complications:

- First, because each entry in a list begins with [->Indent<-][Mar Rel], the sorter reads the text as being in Field 3.
- Second, unless you do something to prevent it, your [List Def] code will get swept up in the sort and perhaps end up in the middle of the list, producing weird results the next time you generate.
- Third, if you use Line Sort, but you have some multi-line entries in the list, the sort will produce hash; so you have to use a Paragraph Sort, but paragraphs in single space are defined by two [HRT]'s, and a List has only a single [HRT] at the end of each item.
- Fourth, if you have multi-line entries in your list, you may want them separated by an extra blank line.

LSTSRT.WPM works right out of the box, provided you mark your list with ASCII 182 codes (hold down Alt and press 182 on the numeric keypad) as set forth above. The included AFF.50 is a document with a list of exhibits at the end, nicely pre-scrambled and marked with ASCII 182 codes, which you can use for a demonstration, and to see how to do the marking. If you want to use some other code, edit the macro to replace each instance of ASCII 182 with your own code. If you want to mark more than one list you'll want to mark each list with a different unique ASCII code, and have a different macro – LSTSRT1, LSTSRT2, and so on – for each list. If you're adventurous, you could even set up a huge chain macro that would look for each of 10 different ASCII codes, and on finding them, sort whatever list is between them. LSTSRT.WPM points the way. Good luck if you try to figure out why I did what I did to make it work.

NOTE: My defaults are set to beep at every opportunity. For this macro, I silenced the beep on a “not found” condition, and turned it back on again at the end. Thus, your machine will emerge beeping on not found. If you don't want that, edit out the string in {LABEL}B~ that reads {Setup}513y{Exit}.

Anders R. Sterner
TPF&S Consulting
June 9, 1988