

## **Manual Pages From Header files**

### **Who is this for ?**

This is a tool for NeXTSTEP developers who need to document their class libraries but have better things to do with their lives than struggle with Edit for days on end.

### **What is it for ?**

The application can be used to generate skeleton manual pages for Objective-C classes from the header files. The man pages are in rich text format with roughly appropriate fonts and layout. The program is not telepathic and will therefore leave gaps in the manual entry for somebody to

fill in. If you hand the output of this program on to a manager without having filled in all of the gaps, expect to be asked to change it ! This is a labour saving tool not a labour replacing utility.

### **Usage**

Use the Document->Open menu to select an Objective-C header file. Ordinary C header files cannot be processed. Normally the header file will be used to generate a rich text format manual page which is written to a file. The Edit application will then be invoked on this file to show the user what has been created. There is an option in the preferences panel to stop Edit from being invoked. The preferences panel can also be used to choose a directory in which to save the completed manual pages. If the application is installed in your ~/Apps directory you should be able to <command. drag-and-drop header file icons from the browser and have the manual page created.

## **Possible extensions**

This tool should be invoked as a service. I have no time to do this today.

All of the headings in the manual pages are taken from an instance of NXStringTable in the nib. Non English speakers can easily change these to produce more readable man pages.

It would be good to be able to apply this filter to a set of file names at once, say the contents of an include directory.

## **Bugs**

The code which parses the header file is in Translator.m. I am not proud of this stuff. In fact I am ashamed of it.

The parsing is not very clever. For instance if structures are defined in the header file, the list of instance variables may be incorrect.

None of the methods are presented in alphabetical order. A routine should be added to sort the strings in the hash tables into alphabetical order before they are written out as rich text.

The rules describing the format of Objective C header files are incompletely followed. For instance if a class adopts any protocols the output will be wrong. Probably this whole tool should be rebuilt around a lex program with the header file format properly defined.

## **I stole some stuff**

The RTF object was stolen from the useful "NeXTSTEP

Programming" book written by the generous and tolerant  
Simpson Garfinkel and Michael Mahoney,  
ISBN 0-387-97884-4, Telos.

### **Disclaimer**

The motivation for writing this code was to produce a tool for my  
own use. I do not suggest that it is bug free and accept  
no responsibility for any badly formatted manual pages produced.  
I am happy for people to change any part of it as long as I,  
Guy Roberts of Object Skills Ltd am accredited.

May 10 1993