Important Release Notes

This is the MiscKit version 1.2.6. It is recommended that you place this folder (ªMiscKitº) in /LocalDeveloper/Source so that it is available to everyone who uses your system. The installed MiscKit will require approximately 15 Megabytes. You will need a little over 3 Megabytes in addition to that if you plan to keep the source code. In order to do a successful ªmake installº from the MiscKit distribution, you will need at least 50 Megabytes free on your hard disk. (65 if you are doing Intel/Motorola/HP.) To build everything at once, including the examples, and then do an install requires approximately 77 Megabytes. (Youch!) You should start the build and then go to lunch...it takes over an hour to build fat. The installation target no longer attempts to build the debug libraries. The new debug target will build and install the debug libraries for you if you need them.

You should be running NEXTSTEP 3.1 or 3.2; the MiscKit is now built fat (m68k and i386) by default. Please see the note below on the current wrinkles involved in changing the target architectures, especially if you are running NEXTSTEP 3.0 or want to use the kit on the HP-PA pre-release. The procedure for specifying compilation architectures has been simplified.

If you use the MiscKit on the HP-PA pre-release, do ***not*** attempt to load the MiscSoundPalette into Interface Builder! It will cause the machine to lock up instantly and you'll have to yank the power cord to bring it back to sanity.

There are audio alerts initiated by the Makefile to encourage you to read this file. As shipped, the Makefile is configured to play them. If you haven't started the build yet then you can disable these alerts by simply changing the AUDIOALERTS variable at the start of the Makefile from ªYESº to ªNOº. The variable SHOWREADME controls whether or not you are forced to see this file.

The Temp folder contains works in progress which are not yet fully integrated into the kit, but may be useful to you nonetheless. These items are documented at the end of the file Authors.rtf. The MOKit is in the process of merging with the MiscKit. The current MOKit is available from the regular NeXT archives.

Originally, it was intended to use Project Builder's ªnewº library project type to maintain the MiscKit. However, Project Builder in 3.2 does not yet have this capability, sadly enough. Since it is possible to use a hacked preamble/postamble file in conjunction with the PB ªtoolº project, we will move to PB control in version 1.3.0.

ExtendedApp is being phased out. The object now adds nothing to the Application class; you should revert to using the Application class. All the methods previously defined by ExtendedApp are now in a category of Application and therefore available to any Application object instance or subclass. The 1.3.0 MiscKit release will not have the ExtendedApp object defined in it, so start getting rid of references to it now; old apps with ExtendedApp in them will still compile, but only while the compatability object remains in the MiscKit.

There are many soft links in the MiscKit. These are required mostly to allow the kit to compile in place before it is actually installed. From the WorkSpace, this will make some directories seem to be infinitely deep. The upshot of this is that (1) if you see something weird, it is probably there for a reason and (2) when copying the MiscKit be sure to duplicate the links rather than copying through the links.

The MiscTee palette will generate warnings (ªstatic access to object of type `id'º) on the MiscTee.m file in three places. These can safely be ignored; they are caused by a hack which is being done in the Obj-C runtime system.

If you find any bugs or have any problems, please let us know!

If you are priniting hard copies of the documentation, the next section (until the part on feedback) is excerpted from the supplied MiscKit Documentation/ directory, so you might want to avoid printint it out twice.

How to install the MiscKit

Basic installation

It is recommended that you place the MiscKit source in /LocalDeveloper/Source. To install the MiscKit, simply type ªmake installº at the MiscKit top level. This will build the MiscKit and install it. In order for the installation to proceed correctly, you should be root while installing the MiscKit. Other available Makefile targets are:

examples: Builds the example apps; this is not done by the install process.
lib: Builds just the MiscKit library.
bundles: Builds the MiscKit bundle projects.
palettes: Builds the MiscKit InterfaceBuilder palettes.
debug: Builds and installs the MiscKit debug libraries.
all: Builds the five targets listed above, without installing them.

uninstall: Removes the MiscKit installation from your system.
distclean: Cleans all the subprojects (examples, bundles, libraries, and palettes) in the MiscKit.

Using the MiscKit libraries

The MiscKit library is installed into /LocalDeveloper/Libraries which is not in the compiler's default search path for libraries. To add this directory to the compiler's search path, add this line to your project's Makefile.preamble:

```
OTHER_LDFLAGS = -L/LocalDeveloper/Libraries
```

Important Note: When you link against the MiscKit library, **don't forget to use the -ObjC linker flag**, which makes sure that all Objects and Categories are linked into your application. If you forget, you will get runtime errors which will crash your application whenever calling a method located in a category. If you are having trouble with the MiscString methods, this is the most likely reason for it.

You can alternatively link against libmisckit.a or libmisckit_g.a, which are links placed in /usr/local/lib, if you prefer, and you won't need to add the OTHER_LDFLAGS above. However, if you use the palettes, the necessary libraries are only installed in /LocalDeveloper/Libraries, so you must either link them to counterparts in /usr/local/lib or add the OTHER_LDFLAGS as above. The headers to go with the palettes are in <misckit/x.h> but are not included by <misckit/misckit.h>. You should include the headers you need.

If you cannot get root permission on the machine you are using, you can still install and use the MiscKit, but you have to be a little bit more resourceful. First, edit the Makefile at the top level of the MiscKit and change the ROOT variable to the full path to your home directory. The MiscKit will install into your account in ~/LocalDeveloper and ~/usr/local/lib. Now, if you so desire, you can move the files around to where you prefer them to be, if it is different from where they are installed. (If you move the files, the uninstall target will not work, however.) In order to use the headers and libraries from their locations within your account, you will need to add a line like this to your project's Makefile.preamble so that the compiler can find them:

```
OTHER_CFLAGS = -I~/LocalDeveloper/Headers
OTHER_LDFLAGS = -L~/LocalDeveloper/Libraries
```

Once you build and install the MiscKit, you can delete the source if you wish, since everything you need in order to use the MiscKit is installed into /LocalDeveloper.

If you have troubles with some of the macros defined in the <misckit/FREE.h> or <misckit/SELECT.h> headers (they don't have the ªMISCº prefix so could possibly clash with your own favorite macros) you can define MISC_SKIP_FREE or MISC_SKIP_SELECT to skip those headers when importing <misckit/MiscBase.h> or <misckit/misckit.h>. Those flags will cause the inclusion of the offending header to be skipped. (If you define these flags you won't be able to use the precompiled headers, though, unless you define the flags when precompiling them. This compiler will give you safely ignorable warnings if it is unable to use ther precompiled header.)

Building thin and fat versions of the MiscKit

The MiscKit by default builds fat, supporting the Intel and Motorola architectures. If you need to build a thin version of the MiscKit or otherwise change the target architectures, you will need to edit the top level Makefile. There are flags for each architecture (MOTOROLA_ARCH, INTEL_ARCH, HP_ARCH, etc.). All you need to do is comment out the architectures you don't want and uncomment the architectures that you do want.

Installing on a NEXTSTEP 3.0 system

If you are running NEXTSTEP 3.0: The MiscKit is developed to run on the most recent version of NEXTSTEP. It should compile without difficulty on either a 3.1 or a 3.2 machine. However, there are a few minor things you will have to do to get the MiscKit to build and install on a 3.0 machine. Since I do not myself have a 3.0 machine anymoreÐall my machines have been upgraded to 3.2ÐI cannot test these fixes, and am therefore reluctant to put them in the Makefiles. According to Steve Hayman and Carl Lindberg you need to do the following to compile for 3.0:

- The MiscKit attempts to compile a fat library, which 3.0 cannot do. Because of this, you currently will have to edit the Makefile in both the top level and the Source directory. In the Source directory, remove the extraneous -arch flag in COMMONCFLAGS. Change ARCHITECTURES in the top level Makefile likewise to affect the palettes and other projects. You should also edit the file Makefiles/lib/Makefile.programs and change the ARCHIVER program at the end of the file to be ªarº instead of ªlibtoolº.

- In regexpr.c, change the declarations of malloc() and realloc() to have a return type of void * instead of char *.

- The top level Makefile attempts to index the installed documentation for Digital Librarian. In 3.0, this crashes, so you should remove the ixbuild call from the top level Makefile.

- In the Palettes, you will need to create the correct directories for the object files from the subprojects. For example, create the directory Palettes/MiscProgressView/obj/MiscProgressView.subproj before compiling the ProgressView palette. Any palette with a subproj in it needs this treatment.

- Also, the palettes which generate libraries such as the MiscProgressView need to have their Makefiles adjusted so that they use the ªarº command instead of ªlibtool.º ªlibtoolº is only in NEXTSTEP 3.1 or higher, and is used so that fat libraries may be generated. Simply uncomment the 3.0 lines in Makefile.postamble and comment the 3.1/3.2 lines.

- The MiscCoolButtons palette seems to tickle a bug in 3.0 (either in NEXTSTEP or in Interface Builder). Apparently, when restarting IB after using this palette, you will get ªUncaught exception: NXReadOnlyString does not respond to -replaceWith:º. For the time being, deleting the file ~/.NeXT/defaults.nibd before restarting IB seems to alleviate the problem. We have no idea why this is happening; if you discover the problem let us know so that we can fix it. Under 3.x where x>0, this isn't a problem.

MiscKit Feedback

(and mailing lists)

To suggest improvements in the kit, direct your comments to the original author of the object(s) in question, as listed in Authors.rtf. Be sure to state the version number of the MiscKit and the resource (object, bundle, etc.). To suggest ideas for new objects or general comments for the entire development team, or to get on the team yourself, send mail to the MiscKit mailing list at misckit@byu.edu (for list traffic) or, if you're not yet on the list, send mail to misckit-request@byu.edu (to either get on or off the list; this address will actually forward to Don_Yacktman@byu.edu).

People interested in discussing the document handling architecture that is under construction (based on the MOKit stuff) can ask Don to place them on the miscdoc list. To send messages to the list, mail miscdoc@alaska.et.byu.edu. Don_Yacktman@byu.edu will put you on or off the list. Only request this list if you are seriously interested in participating in the discussion, as there can be bursts of volume on the list and also beware that NeXTMail is allowed on the miscdoc list!

People interested in discussing a reorganization and redesign of the MiscString class can ask Don to place them on the miscstring list. To send messages to the list, mail miscstring@alaska.et.byu.edu. Don_Yacktman@byu.edu will put you on or off the list. Only request this list if you are seriously interested in participating in the discussion, as there can be bursts of volume on the list and also beware that NeXTMail is allowed, and often encouraged, on the miscstring list!

All three discussion lists are archived at ftp.byu.edu:/pub/next/misckit if you wish to view past messages.

Items in the MiscKit will be useful to some, and very lacking for others. The best thing to do is to bug the various authors about features that you need fixed or implemented. Submissions of your own objects are welcome and we also welcome ideas for new objects that you would find useful. The areas most frequently requested will, of course, receive more attention. So if you find any bugs, or have any problems with portions of the MiscKit, please let us know.

Other Notes

In spirit, license and use is utterly free. Commercial, shareware, and freeware apps can all use this stuff as much as they want as long as they comply with the (very few) requirements detailed in the file License.rtf. That also means there's no warranty, no support, and you are on your own. (Of course, there are many helpful folk on the MiscKit mailing list, so support does *exist*, although it is not *promised*.) If you use the MiscKit, you should acknowledge that fact in your app somewhere, but you have no other obligation to meet in order to use the MiscKit.

Despite the lack of official support, your feedback is very important. It would be appreciated if you would send any changes you make in the MiscKit source code to the original author of the modified resource so that others may benefit from your changes.

Share and enjoy![2]

±Don_Yacktman@byu.edu
MiscKit administrator

[2] Douglas Adams, The Hitchhiker's Guide to the Galaxy. :-)