BusyBox

BusyBox is a trivial program which exists to provide on-line guidance about the user interface conventions under NeXTstep.   The code is very simple; BusyBox is not much of a programming example.   You may want to check out the Help object as a possible schema for implementing help in your program. (Applications written for NeXTSTEP 3.0 can also use the help facilities provided in the Application Kit.)   And if you have need for a clock or a gauge, you can look over the ClockView and GaugeView.

I created BusyBox so you could run it to learn more about the user interface guidelines and hopefully have some fun while you're at it!

Classes of this application:

Help   This class encapsulates all the functionality for a help object which can maintain a list of help topics, display the file for any help topic and respond to requests for context-sensitive help. The help object owns its own nib section "Help.nib" which has a

help panel - with an NXBrowser to display the help topics and a scrolling text view to display the help files.   The help files are all stored as RTF text files in a directory called Help within the app wrapper.   At init time, the Help object loads the browser with names of all the files found in the Help directory.   When a nam(W chosen from the browser, the help object opens a stream on that file and read the rich text into the text object. The help object also responds to request for context-sensitive help, by trying to find an appropriate help file for the view that was moused down in.   This object is a useful addition to any program, because all users appreciate help!

BusyBoxApp Mostly, this subclass of Application exists to override sendEvent:. When it catches a Control-mouseDown going through, it vectors it off to the Help object.   Other responsibilities of the BusyBoxApp include loading the Info and Preferences Panels when needed and keeping track of the path to the app wrapper.

AnimalView   Simple subclass of view to display an NXBitmapImageRep. Knows how   to flip and rotate itself in response to IB controls.

ClockView Subclass of view to implement a simple clock.   It can draw an analog face, a digital clock or a sundial.   You have the option of turning the seconds hand on or off, as well as controlling whether the date   is also displayed.

GaugeView Subclass of view to implement a simple round analog gauge. You can set the minimum, maximum value, start angle, angle range, title, font, and more.   It is a pretty generic round gauge view, if you ever have need for one.

Other files:

BusyBox.nib   The main nib file.   Contains the Busy Box window.

InfoPanel.nib   The nib module containing the Info Panel (loaded only on demand).

PrefPanel.nib   The nib module containing the Preferences Panel (loaded when needed).

Help.nib   The nib module owned by the Help object - has the Help Panel, browser of help topics, and the scrolling text view to display the help files.

Clock.psw	pswraps used by the ClockView to draw and animate the clock faces.
Gauge.psw	pswraps used by the GaugeView to draw and animate the analog gauge.
Clock.strings	A string file for NXStringTable, contains names of weekdays and months.
Cow.tiff, Duck.tiff, Horse.tiff, Jaguar.tiff, Pig.tiff	Tiff f(X for the AnimalView.
Makefile.preamble, Makefile.postamble	Additions to the Makefile provided by Interface Builder.
IB.proj, Makefile, BusyBox_main.m, BusyBox.iconheader	These files created by Interface Builder.

Topics of interest from BusyBox:

Overriding sendEvent: to do context-sensitive help (BusyBoxApp)
Filling and using an NXBrowser (Help)

Reading rich text files into a scrolling text view (Help)
Using a timed entry (ClockView)
Simple pswraps (Clock.psw and Gauge.psw)
Creating an app wrapper (Makefile.preamble,Makefile.postamble, IB.proj)
Multiple nib files (BusyBox.nib, InfoPanel.nib, PrefPanel.nib,Help.nib)
Loading a nib module on demand (BusyBoxApp)
Scaling and rotating an NXBitmapImageRep (AnimalView)
Entry validation using text delegate method textWillEnd: (GaugeView)
Constraining the resizing of a window using delegate methods (Help)
Storing and accessing strings using an NXStringTable (ClockView)