

**objectThreadPerform** - Multiple thread support

**ScrollText** - Text object printing support

**ReadConfig** - Run-Time configuration support

by Martin D. Flynn, NeXT Computer, Inc. (*[martin\\_flynn@next.com](mailto:martin_flynn@next.com)*)

Permission is granted to freely redistribute any of the following classes, including the source code, and to use fragments of this code in your own applications if you find them to be useful. These classes, along with the source code, come with no warranty of any kind, and the user assumes all responsibility for its use.

**TestClass.app** is the name of the application that demonstrates the usage of the following classes. See below.

**objectThreadPerform** is a category or Object that provides a general solution for thread support which includes support for allowing threads to communicate with the window/event manager. Since it is a category of Object, all classes have access to the features provided. Using the "forkPerform:detach:" family of methods, a thread can be created to perform some action in the background while the window manager (main thread) is free to handle events. The "mainThreadPerform:wait:" family of methods can then be used by the secondary thread to send messages to be executed by the main thread. Since only the main thread can successfully/efficiently draw to

the screen, this can be used by the secondary thread to cause information to be displayed. See the **objectThreadPerform.h** header file for more usage information.

**ScrollText** is a class that provides a full-featured set of utilities to handle printing to a Text object within a ScrollView. This includes the ability to change the text font, and color, of the text printed to the Text object. This class will also handle text sent to it from any secondary thread. It will automatically send the data to be printed to the main thread to be printed into the Text object (only if **objectThreadPerform** is compiled with the application). See the **ScrollText.h** header file for more usage information.

**ReadConfig** is a class that provides a general solution to applications that require run-time configuration. A configuration file name can be passed to the method "readConfigFile:target:", or "readAppConfig:target:". The file will be parsed and interpreted as method names and arguments to send to the specified target. Methods requiring no-arguments, or one-argument, are allowed. The class also contains some primitive branching controls to provide a simple control language. See the **ReadConfig.h** header file for more usage information.

**TestClass** is a sample application that demonstrates some of the features of the above classes. The application is controlled by the methods listed in the configuration file "**testConfig.cfg**" in the app wrapper. The configuration file instructs the application to print some text to the ScrollView. First from within the main thread, then from a secondary thread. The printing also demonstrates how to

change the colors and fonts of the text to be printed.