DirPanel

CATEGORY OF	SavePanel
INHERITS FROM	SavePanel : Panel : Window : Responder : Object
DECLARED IN	"DirPanel.h"

CATEGORY DESCRIPTION

DirPanel provides the ability to select directories for SavePanels and OpenPanels. It does this by providing an Objective-C Category of SavePanel, and an alternative form of the **-runModal*** methods for selection of directories. To run a directory selection panel, create an appropriate instance of SavePanel or OpenPanel, and call **-dirPanelRunModal** on it. This method returns non-zero for success, or zero on Cancel, just like regular **-runModal*** does. The directory selected may then be retrieved via the **-directory** method.

Though DirPanel makes great effort to retain most similarity to SavePanel and OpenPanel, sometimes it will act slightly differently. I have made every attempt to make such differences err on the side of caution, which means that sometimes you have to apparently select an element twice. This is better than selecting elements that you did not mean to select, though. If you can quantify such a problem in a decent manner, send me email and I will attempt to fix it.

To include DirPanel in your project, simply include it as a class.

NeXTSTEP3.0 is supposed to have a fixed version of SavePanel/OpenPanel that allows the user to accomplish this type of thing in a NeXT-approved manner. When I have access to a NS3.0pr, I will look into generalizing the DirPanel code in such a way that under NS3.0, it will call the NeXT-approved methods, while under NS2.x it will hack them. In that way apps compiled for NS2.x would automatically take advantage of any improvements NeXT has managed in this functionality. [Anyone out there with a NS3.0pr and some time on their hands can feel free to do this and send it to me. The access to NS3.0pr is not the main problem - it's the "time on my hands" part that gets me every time.]

DirPanel is free. Use it and abuse it as you see fit.

Scott Hess
12901 Upton Avenue South, #326
Burnsville, MN 55337
(612) 895-1208
scott@gac.edu
shess@ssesco.com

INSTANCE VARIABLES

Inherited from SavePanel	id	browser;
	id	form;
	id	okButton;
Declared in DirPanel	static BOOL	notCancel;
	static id	realOkButton;

browser — Hacked at to keep more information visible at any one time, and also to provide double-click-ability for directories.

form — Area where textual user input may occur.

okButton — Hacked to allow processing of the current selection, and acceptance of it as needed.

notCancel — Flags when a 'psuedo-Cancel' is used to trick SavePanel into accepting a directory.

realOkButton — Stores the *okButton* while in **-dirPanelRunModal**.

METHOD TYPES

Simulating the *okButton* - forward::
- setEnabled:

Providing directory selection - dirPanelRunModal
± realOk:

INSTANCE METHODS

dirPanelRunModal

-(int)**dirPanelRunModal**

Hacks a SavePanel or SavePanel subclass instance to allow selection of directories, and then does a **-runModal*** on it. Returns YES if there was a successful selection, with the selected directory accessable via **-directory**. Returns NO if the selection was Cancelled.

forward::

- **forward:**(SEL)*aSelector* **:**(marg_list)*argFrame*

Forwards messages to *realOkButton* while in **-dirPanelRunModal**.

realOk:

- **realOk:***sender*

Action which handles clicks on the *okButton* and double-clicks on *browser* while in **-dirPanelRunModal**. If there is a value in the *form*, the data is extracted and loaded into the *browser*. Otherwise, a 'puesdo-Cancel' is executed to exit the modal event loop, with appropriate values set up so that **-dirPanelRunModal** realizes that the Cancel was not real.

setEnabled:

- **setEnabled:**(BOOL)*flag*

Intercepts the *okButton*'s **-setEnabled:** message, and substitutes YES for *flag*.   This causes the *okButton* to remain enabled in all cases.