

3.2 Release Notes: Application Behavior

This file contains release notes for the way applications work under NEXTSTEP Releases 3.2, 3.1, and 3.0. Items specific to or introduced in Release 3.2 are listed first, followed by the 3.1 and 3.0 notes.

Notes Specific to Release 3.2

Changes

Printing Multiple Copies of a Document

When multiple copies are requested in the Print panel, NEXTSTEP effectively creates separate print jobs for each copy. On some printers this can be quite slow. A default has been provided to cause the multiple-copy information to be imbedded

in the print job itself.

WARNING: This default will be ignored in certain cases. This will cause one copy of the job to be printed, no matter how many copies have been specified in the Print panel. Use the default carefully. It is recommended that the default be set for a particular app, and then the printing behavior of that app be tested to make sure it is satisfactory.

- Certain applications don't produce print jobs that include the multiple-copy information. The Preview application does not, for example.
- NeXT printers on hosts running pre-3.0 NEXTSTEP ignore multiple-copy information imbedded in a print job.

To set the default:

```
dwrite appname NXMustPrintCollated NO
```

To reset it:

```
dremove appname NXMustPrintCollated
```

Notes Specific to Release 3.1

Changes

Menus and Pop-Up Lists

When the user presses a button to bring a pop-up list on-screen, the list will now always be aligned so that the currently selected item appears directly over the button. The list no longer adjusts its position automatically so that all the items are visible.

However, both pop-up lists and menus can be made to reveal items which extend beyond the edges of the screen. If some of a menu or list is off-screen and the user drags towards the edge of the screen, more and more of the menu or list will slide into view. This feature is similar to autoscrolling a document. To use it, simply drag the cursor towards the edge of the screen while performing a selection within the menu or pop-up list; once the cursor is within two pixels of the edge, the menu or list will start sliding to reveal more of itself. This feature is especially useful on smaller screens.

New Open Panel and Save Panel Buttons

The Home button in an Open or Save panel now displays a small icon, the default image for the user's home directory, rather than the word "Home." Its function is still the same—it selects the user's home directory.

A new Disk button and Disk Eject button now appear alongside the Home button.

The Disk button displays the image of a floppy disk and, when clicked, selects the directory of a currently mounted removable disk. If more than one removable disk is mounted, each click will select another disk directory; directories are taken in alphabetical order. If a disk is inserted but not mounted, clicking the Disk button will mount it. (See the new method **getMountedRemovableMedia:** discussed below.) The Disk Eject button displays the image of a floppy disk with an arrow pointing down and, when clicked, ejects the currently selected floppy (or the only floppy if a single disk is mounted). **It is very important that the user either press this eject button or use the Eject command in Workspace Manager before physically ejecting a disk.**

CMYK Colors

The 3.1 Window Server has the ability to use an enhanced algorithm for rendering device-dependent CMYK colors on an RGB device. This results in greater accuracy in the on-screen appearance of CMYK colors and can be especially important in publishing applications.

In 3.1, this feature is disabled by default, as it tends to make a lot of older applications behave strangely. It may be enabled for an application by setting the **NXCMYKAdjust** default to YES:

```
dwrite appname NXCMYKAdjust YES
```

Specifying GLOBAL as the *appname* will enable it for all applications. One

application for which it makes sense to enable this feature is Preview. It permits Preview to display CMYK colors much more like they will appear when printed.

Bug Fixes

The following bugs are among those that have been fixed in the 3.1 version of the Application Kit:

User-Level Bugs

Reference	16224
Problem	Security problem with /usr/bin/copy and /usr/bin/paste .
Description	In 3.0, it was possible to log in to a remote machine (rlogin) and using the copy and paste command-line utilities read from and write to the pasteboard on that machine. In 3.1, this is possible only if the user who started up NEXTSTEP on the remote machine has the Public Window Server preference enabled. Otherwise, the copy and paste utilities still work, but the pasteboard used by NEXTSTEP applications running on the machine is protected from remote access.
Reference	25576
Problem	A document with data linked to another document might sometimes

	be displayed to the user as having unsaved changes, when it didn't in fact.
Description	Saving a source document could cause a document linked to the source to have out-of-date information about the link. Consequently, when the file with linked data was opened again, it might have been displayed with the "unsaved changes" close button.
Reference	25587
Problem	Opening certain TIFF files and compressed files could cause an attention panel to appear notifying the user that "Workspace Internal Error 1000 occurred."
Description	The internal error should no longer occur and the panel should no longer appear.
Reference	27998
Problem	Printing certain dictionary entries from the Webster application failed.
Description	Because of a bug in the Text class code that handles diacriticals, certain entries couldn't be printed directly from the Digital Webster application.

Reference	28895
Problem	A * Library/Fonts directory couldn't contain more than about 250 fonts.
Description	A bug in buildafmdir caused it to stop working after about 250 fonts. Since most font families have several different fonts (Times-Roman, Times-Bold, Times-BoldItalic, etc.), this meant that you couldn't have more than 50 or 60 different kinds of fonts.
Reference	28994
Problem	JPEG compression periodically mishandled colors.
Description	JPEG sometimes converted the red channel to green.
Reference	29482
Problem	Clicking the Forget button in a Spelling panel sometimes caused a crash.
Description	If the user checked the spelling in a document, closed the document (while another was still open on-screen), then clicked the Forget button, the application would crash.

Reference	29969
Problem	Some JPEG-compressed TIFF images failed to load.
Description	Some TIFF images created under releases prior to 3.0 and compressed using JPEG could not be loaded and displayed.

Reference	29979
Problem	The rectangles that animate miniaturization and deminiaturization were sometimes displayed within the deminiaturized window.
Description	After a window was deminiaturized, it might contain bits and pieces of the rectangles that the Application Kit used to animate the miniaturization and deminiaturization processes. This happened only for certain applications.

Reference	30279
Problem	Beginning with Release 3.0, the Public Window Server option in Preferences must be enabled for applications on other machines to connect to a local application.
Description	This change adversely affected applications (such as network games and network license servers) that require interapplication

connections between machines. In 3.1, the change will not affect applications built under releases prior to 3.0. See the section on "Secure Ports" above.

Reference	30376
Problem	If the ImageAgent filtering service was installed, some images in ".rftd" directories appeared as the NeXT logo.
Description	Because of a bug in the way the Text class handles ".rftd" directories, image data in files with names less than eight characters weren't imaged. Instead, the default image, a NeXT logo, were displayed.
Reference	30599
Problem	AdobeGaramond and ITC Garamond fonts could not coexist in the Font panel.
Description	When both font families were installed, the Font panel would only display AdobeGaramond; none of the ITC Garamond faces were listed.
Reference	30908

Problem	A data link that was deleted and then reestablished (through the Undo command) didn't reflect changes made to the source document in the interim.
Description	If the user deleted a data link, edited the source document, then undid the deletion to reestablish the link, the data displayed would be the same as before the deletion. It wouldn't reflect the changes made to the source document during the period when the link was deleted.
Reference	30910
Problem	When the system lost track of a data link connection and the user again identified the source document, it wouldn't check to see if the reestablished link needed to update from the new source.
Description	When the user repairs a broken link, the system wouldn't automatically make sure that the linked data was up-to-date.
Reference	30973
Problem	Drawing images from EPS files would take a long time (every 12th image).
Description	A new drawing context was created and destroyed for each image

drawn. This would trigger garbage collection every twelfth context

Notes Specific to Release 3.0

These notes were included with the Release 3.0 version of the Application Kit. Sections that are no longer relevant have been marked with an italicized comment.

Changes

- RGB bitmap images, EPS images, and non-CMYK spot colors are printed on PostScript Level II printers using a calibrated color space; this should provide better color matching on Level II printers. In addition, print output of old applications (applications built on a pre-3.0 NEXTSTEP system) is also color calibrated by interpreting all device-dependent color operators (except for those specifying CMYK colors) as calibrated. This is done because the chances that these old applications really wanted device-dependent colors is small.

For the most part, all this means is that users will now get color calibrated output from imported images when printing to Level II printers. However, under this scheme there is a chance that some EPS files will print out wrong. The most likely problem is that some colors will be incorrect (not just off, or badly calibrated, but incorrect). We don't expect this to be a problem for any EPS files generated by apps on the NeXT machine. However, if you do encounter such an EPS image, the way to get it to import correctly is to disable this forced color calibration in the

application you're using. This can be accomplished in a Terminal window by:

```
dwrite appname NXColorCalibrateLevelOneOps NO
```

This will turn off all forced calibration of device-dependent operators in that application. It will not disable color calibration in kit objects manipulating spot colors or bitmap images; thus colors chosen from the color panel and TIFF images will still print out calibrated. If you encounter an application where the spot colors or TIFF images are displaying or printing incorrectly, you can turn off all generation of calibrated colors in an application by:

```
dwrite appname NXUseCalibratedColor NO
```

If you encounter a shipping application which works incorrectly without this `dwrite`, please report this as a bug.

- Colors can now be dragged between wells (and other color receptors) in different applications.
- The FontPanel and Services menu will now reflect any added Fonts or Services (respectively) when they are ordered front by the user. Thus, you no longer have to quit your application to take advantage of new fonts or new services, just close and open the affected UI element (the FontPanel or Services menu).
- The FAX Cover Sheet format has changed, so your existing cover sheet template will not work. The **Draw** application in **/NextDeveloper/Demos** can be used to

create cover sheets.

- Submenus which were torn off at the time an application quits will be restored in the same locations when the application is restarted. The location of the main menu, if the main menu has been moved, is also remembered on an application by application basis.
- Miniwindows now reuse vacated spots. If a window has been miniaturized at any time in the past (even if it is not currently miniaturized) the slot it occupies will not be reused until it is closed. Miniwindows will also now crawl all the way to the top of the screen instead of overlapping each other. Window miniaturization is also combined with some animation to indicate where the miniwindow ends up.
- Panels which normally only become key only if needed can be made key by clicking on their title bar (moving the window will NOT make it key, you must click).
- The FontPanel, when key, will select in the Family scrolling list when you type the first few characters of a family name.
- You can now set the color of a selected range of text in the Text object by dragging a color onto the selection.
- Using arrow keys in the Open/SavePanel browser has been made to operate like the arrow keys in the WSM's browser. You must hold down the Control key for the right and left ones to work since the TextField wants them too.

- The return-sign icon on AppKit panels will be removed when the panel stops being key. This should become part of our UI guidelines since it is somewhat confusing for people to see that icon, hit return, and not get what they expect because the window with the button was not key.
- Shift-clicking on the Preview button in the FontPanel will cause the Preview button to stay "on" and automatically preview fonts as you click in the panel.
- The width of columns in the Open/SavePanels is now linked to the one in the Workspace (an vice versa). You can resize your Open/SavePanels without adding columns to their browsers by holding down the Alernate key while resizing.
- Title bars showing filenames use an ellipses to abbreviate the directory name if the whole "file ± directory" title can't fit.
- Windows which are too big for the screen will now be resized smaller if they are resizable and are allowed (programmatically) to change size. This allows for apps with large windows to run on smaller screens. Window layouts which have been designed in Interface Builder will also attempt to look decent when instantiated on screens of different sizes by trying to preserve the ratio of the distances on all sides. (Interface Builder in 3.0 provides a window autopositioning panel to provide developers with more explicit control over window placement.)
- By default, Cell selection in Matrices now takes place based on rectangular

regions, not by rows

- When a disabled cell in a Matrix is clicked on, other cells which may be highlighted will no longer become unhighlighted.
- Shift+mouse down/drag did not used to highlight any cells if there wasn't already at least one cell highlighted. This has been corrected. If there are no cells highlighted, and shift+mouse is pressed/dragged, the behavior will be identical to the behavior if the shift key was not being held down. The same holds true for Alt+mouse down/drag.
- You can now specify the time you want a Fax to be sent. This can be used to delay sending a Fax until late at night, for example, when the phone usage rates are low.
- The PrintPanel has been changed to build the Paper Feed and Resolution buttons based on the PPD info in the printer in the PrintInfo. If the PPD info is not available, a message is shown rather than the buttons. Otherwise, the available options are shown on the buttons. Also, a new button called "Options" has been added, which brings up a panel containing all printer features that can be determined from the PPD file. This button is not enabled unless features besides Paper Feed and resolution are found. If there is more than one input tray, two Paper Feed options are displayed in addition to the input trays: "Any Tray", which tries to select an appropriate tray, and "Split" which uses the *second* input tray listed for the first page of the job, and the first input tray for the rest of the job.

- The PrintPanel notices when a user specifies page numbers for a job in reverse (pages 5 through 1). This used to be an error. Now, it reverses the page order of the generated print job using `setReversePageOrder:` in PrintInfo.
- It's not possible to use `-NXHost` from a machine running NEXTSTEP 3.0 to a machine running an earlier release. However you can run an app on a NEXTSTEP 2.x machine and have its windows come up on a NEXTSTEP 3.0 machine.
- NEXTSTEP 3.0 supports draft 1 of revision 6.0 of the TIFF standard, published Feb 14, 1992. Because this revision provides specifications for writing JPEG images and including transparency information in TIFF images, JPEG images and images with alpha as written out by NEXTSTEP 3.0 should be compatible with other systems. However, JPEG images written out by 3.0 cannot be read back on 2.0 systems. Images with alpha are still interchangeable between 2.0 and 3.0 systems.

12-bit grayscale TIFFs can now be read in, written out, and displayed.

The command line program `tiffutil` has been updated to support the new features, including JPEG compression.

- The ColorPanel user interface has been redesigned.
- User color lists are now stored in `~/Library/Colors`. 2.x color lists (which lived in `~/NEXT/Colors`) will still be read in and displayed by the color panel;

however, when saved, they will get saved in the new location, `~/Library/Colors`. Users who upgrade to 3.0 and don't plan on using 2.x may remove the `.color` and `.tiff` files from `~/NeXT/Colors`.

- The custom color list mode of the Color Panel now provides PANTONE colors. PANTONE color tables are provided for some PostScript printers and CMYK values from these tables will be used when printing to such a printer. If a device-specific table isn't available, then the color simulation CMYK values for 150-line imagesetting will be used.

Note that if you use PANTONE colors in your documents, the documents will most likely fail to load properly on any release prior to 3.0. Please keep this in mind in creating documents that need to be displayed on 2.x machines. If you create PostScript files from such documents, however, the documents will display and print fine under NEXTSTEP 2.x.

A default has been provided to assure that an application under 3.0 will not write out PANTONE colors. This default maybe used to "fix" 3.0 files: Quit the application if running, set the default, load and save the files under 3.0, quit the app, and then reset the default. *Warning:* The saved files will have no PANTONE colors; they will be replaced by the RGB values of the PANTONE colors for the monitor. This default is provided "just in case" and should be used carefully, only to create copies of documents that will load under 2.0.

To set the default:

```
dwrite appname NXSave2.0Compatibly YES
```

To reset it:

```
dremove appname NXSave2.0Compatibly
```