MENU_NROWS attribute has been implemented (in previous releases this attribute existed didn’t work as expected)

notifier errors are no longer generated when application is abnormally terminated via “kill <pid>”

Notices flush the input queue before being displayed (this prevents them from being accidently dismissed when the user enters a CR before the notice is displayed)

‘tip’ now works in a cmdtool

XView no longer grabs the select button if running in “FollowMouse” mode

password security hole plugged

overlapping cmdtools and shelltool are properly redrawn after scrolling

text caret is moved correctly when navigated with arrow keys inside a panel text item

Important Bug Fixes

Numerous memory leaks have been plugged in menu, panel and frame creation/deletion. Additionally, many textsw and cmdtool memory leaks no longer exist.

cmdtool -M (limits size of cmdtool tmp file) has been enhanced so cmdtool memory size doesn’t grow unchecked and unbounded

Time required to create, delete or modify a panel text item has been significantly reduced (particularily noticeable when managing many text items).

CR no longer turns into a LF when running a program in a shelltool that reads text in ‘raw’ mode.

panel text and numeric text items have auto-scroll buttons

multiple column menu implementation has improved

TEXTSW_ACCELERATE_MENUS

Controls whether the TEXTSW menus are accelerated. Create only attribute, default is False.

INTERNATIONALIZATION Support:

Wide Character API
Multiple Input Methods
Support for font sets.
Single library for domestic and international release.

PANEL_LIST_DO_DBL_CLICK

Informs the panel list to interpret two select events that occur within the OpenWindows.MulticlickTimeout value, as a double-click instead of as a second select or a deselect. PANEL_LIST_OP_DBL_CLICK will be delivered.

PANEL_LIST_ROW_VALUES

A new high performance insertion and retrival method for the PANEL_LIST package. Allows batch insertion and retrieval of list rows.

PANEL_LIST_MASK_GLYPH
PANEL_LIST_MASK_GLYPHS

Specifies a Server_image to be used as a clip-mask for the corresponding glyph.

New X Default: “Text.DeleteReplacesClipboard”

Controls whether pending delete operations replace the contents of the CLIPBOARD. Default is False.

Other New Features

PANEL_BORDER

Adds Open Look compliant 3D chiseled border to a Panel. Single line border is 2D.

PANEL_POST_EVENTS

Assists in the use of the Notifier’s Interposition model with Panel Items.

PANEL_LIST_INACTIVE

“Grays out” a row in a panel list.

PANEI_LIST_DELETE_INACTIVE_ROWS

Deletes all inactive rows from the panel list (similar to PANEL_LIST_DELETE_SELECTED_ROWS).

File Chooser (continued)

- Consistent file chooser (look and feel) available on the desktop in 4/93.
- XView and OLIT’s file chooser will have a similar look and feel.
- Extensive API available to control attributes of the file chooser.

- Three objects comprise the XView implementation of the file chooser:

File List:	Subclassed from the PANEL_LIST, this object provides display and navigation of directories in the file system.
Path Name:	Subclassed from the PANEL_TEXT object, this object handles expansion of UNIX path names
History:	This object provides for the sharing of data amongst several objects. This satisfies the need to share a list of common information across several instances of the file chooser within an application.

And the new way...

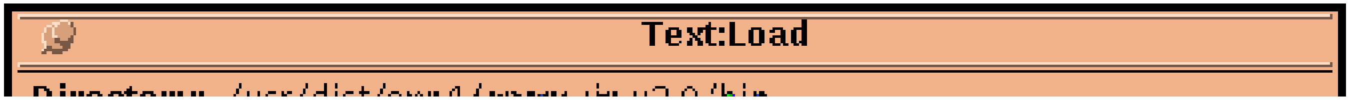
Save

Open

Type in the path to the folder and press Return

File Chooser

The old way...



Procedural flow difference

The procedural flow taken when a menu is brought up using the MENU button is different from when a menu accelerator is used.

Menus are normally brought up (when the user presses the MENU button) in event callbacks :

```
window_event_proc(window, event)
Xv_window     window;
Event         *event;
{      /* look for menu button event */
       if (event_action(event ) == ACTION_MENU &&
              (event_is_down(event)) {
             menu_show(menu); /* display menu */
       }
}
```

Menus are not brought up when menu accelerators are used; the menu item callback procedures are called directly.

There is a notion of a ‘‘Core Set’’ of accelerators.

It is a set of commonly used accelerators such as save, print and quit.

Core Set accelerator bindings are defined in:
$OPENWINHOME/lib/locale/<locale>/xview/defaults.

The XView API uses an abstraction to bind the accelerator to the core set:

```
save_item = xv_create(NULL, MENUITEM,
                 MENU_STRING, “save”,
                 MENU_ACCELERATOR, “coreset Save”,
                 MENU_NOTIFY_PROC, save_action,
                 NULL);
```

The X resource entry for core set accelerators is different from the non-core set case:

```
OpenWindows.MenuAccelerator.Save:          Meta <Key>s
OpenWindows.MenuAccelerator.Quit:         q+Meta
foo.load.accelerator:                      Meta+b
```

```
load_item = xv_create(NULL, MENUITEM,
                  XV_INSTANCE_NAME, “load”,
                  MENU_STRING, “load”,
                  MENU_ACCELERATOR, “Meta+l”,
                  MENU_NOTIFY_PROC, load_action,
                  NULL);
```

“Meta+l” can be overriden by having the following entry in your X resource database:

foo.load.accelerator: Meta+b

The load action will be done when “Meta+b” is pressed instead of “Meta+l”.

Note: XV_INSTANCE_NAME will need to be used to give the menu item an instance name.

Customization of menu accelerators

- Can be done using the X resource database

- Overrides the attribute interface

- The resource name used has the format:

<app>.<menu item>.accelerator:<accelerator string>

<app>	application instance name
<menu item>	menu item instance name (XV_INSTANCE_NAME)
<accelerator string>	Accelerator Binding Syntax

e.g.

```
mailtool.compose_new.accelerator: Meta+N
```

Example: In the application ‘foo’, if we have:

Menu Accelerators

Menu attributes

- Attributes that specify accelerators for menu items.
- Corresponds to a currently existing method of creating a menu item.

MENU_ACCELERATOR
MENU_ACTION_ACCELERATOR
MENU_STRINGS_AND_ACCELERATORS

Frame attributes

- Attributes to associate menus with frames.
- Necessary to inform frame what accelerators on menu to detect.

FRAME_MENUS
FRAME_MENU_ADD
FRAME_MENU_DELETE
FRAME_MENU_COUNT

XView 3.2 - Update

- Menu Accelerators
- File Chooser
- Other New Features
- Important Bug Fixes