The Main Menu

The main menu works best when it's short (so that commands are easy to find) and narrow (so that it doesn't take up much screen space). Applications should try to have not many more than eleven or twelve commands in the main menu. (Nine commands are shown for the standard menu, and only three of themÐInfo, Hide, and QuitÐare required of all applications.)

The main menu is also, for'i most part, a menu of menus. Commands that are added to the main menu should typically be ones that bring up submenus. When designing your user interface, you can move a command that the guidelines place in a submenu up one level to the main menu, provided that:

- The main menu is short enough to accommodate another command (see the guideline above).

- The command provides functionality that's considered central, even crucial, to the application. For example, a text editor might bring the Font command up to the main menu from the Format menu, but a spreadsheet would not.

Like any other command that's added to the main menu, a command that's raised from a submenu should generally control another submenu. When a command is promoted to the main menu, it should, for continuity, be located immediately after the command for the submenu it would otherwise be in. For example, if the Font command is raised from the Format menu, it would logically follow the Format command. If the Find command is promoted from the Edit menu, it follows the Edit command.

There's a great deal to be gained if commands shared by most applications are arranged similarly in similar menus. Every application should strive to lay out its main menu like the model illustrated by BusyBox.

Command	**Action**
Info	Attaches the Info menu, which contains commands for application information, preferences, and help. If an application doesn't support any of the commands in the Info menu except Info Panel, it should dispense with the menu and make the Info command in the main menu bring up the panel instead.
Document	Attaches the Document menu, which contains commands affecting windows and the files displayed in windows. The Document menu is a menu that will be named differently in different applications. Therefore, it's important that the command be in a prominent, well-defined location (second).
Edit	Attaches the 'p menu, which contains commands affecting the current selection in editable documents.
Format	Attaches the Format menu, which contains the principal formatting commands for the application.
Windows	Attaches theWindows menu, which contains commands to bring windows to the front of the screen and to close and miniaturize windows.
Print...	Brings up a panel that permits the user to print a document. This panel is defined in the

Application Kit.

Services

Brings up a menu with commands that request services from other applications. For example, users can request the dictionary application (Webster) to look up the current selection, or the Workspace Manager to open a file. The Services menu will become increasingly important over time and therefore is placed in an easy-to-find position (just above Hide). It's not higher in the main menu because it isn't a gateway into the application's own functionality (but often invokes a systemwide service or activates another application).

Hide

Hides all the windows of the application.

Quit

Terminates the application. If the application has any open files that the user has altered but not saved, this command should bring up an attention panel giving the user the option of canceling the command and perhaps also of saving the altered files.