Subsections

The Common Desktop Environment

In order to use Solaris and most other Unix Systems you will need to be familiar with the Common Desktop Environment (CDE). Before embarking on learning C with briefly introduce the main features of the CDE.

Most major Unix vendors now provide the CDE as standard. Consequently, most users of the X Window system will now be exposed to the CDE. Indeed, continuing trends in the development of Motif and CDE will probably lead to a convergence of these technologies in the near future. This section highlights the key features of the CDE from a Users perspective.

Upon login, the user is presented with the CDE Desktop (Fig. 1.1). The desktop includes a front panel (Fig. 1.2) , multiple virtual workspaces, and window management. CDE supports the running of applications from a file manager, from an application manager and from the front panel. Each of the subcomponents of the desktop are described below.

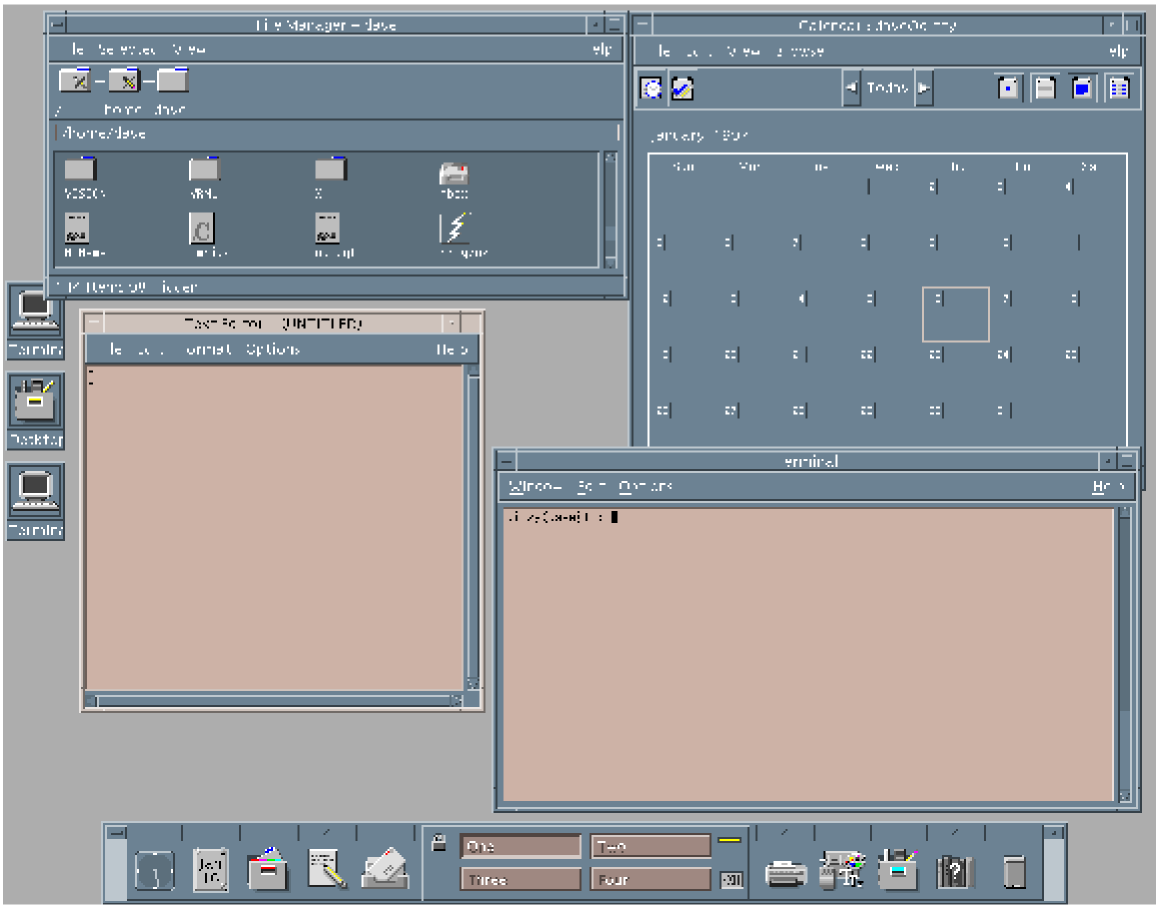


Fig. 1.1 Sample CDE Desktop

The front panel

The front panel (Fig. 1.2) contains a set of icons and popup menus (more like roll-up menus) that appear at the bottom of the screen, by default (Fig. 1.1). The front panel contains the most regularly used applications and tools for managing the workspace. Users can drag-and-drop application icons from the file manager or application manager to the popups for addition of the application(s) to the associated menu. The user can also manipulate the default actions and icons for the popups. The front panel can be locked so that users can't change it. A user can configure several virtual workspaces -- each with different backgrounds and colors if desired. Each workspace can have any number of applications running in it. An application can be set to appear in one, more than one, or all workspaces simultaneously.

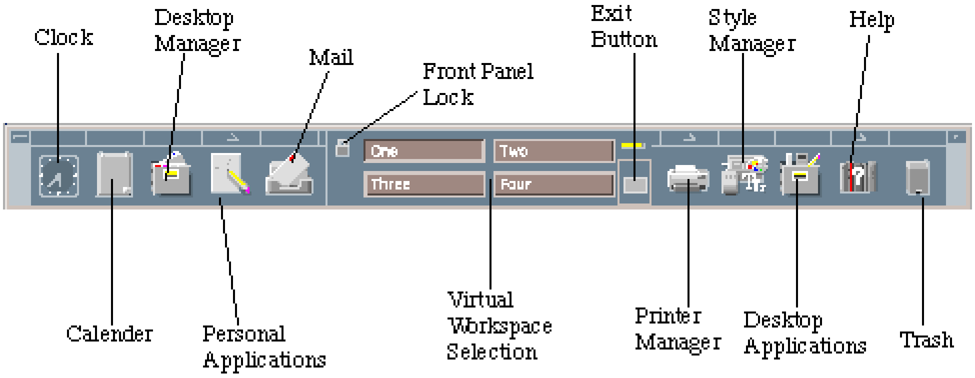


Fig. 1.2 Clients, Servers and Xlib

The file manager

CDE includes a standard file manager. The functionality is similar to that of the Microsoft Windows, Macintosh, or Sun Open Look file manager. Users can directly manipulate icons associated with UNIX files, drag-and-drop them, and launch associated applications.

The application manager

The user interaction with the application manager is similar to the file manager except that is is intended to be a list of executable modules available to a particular user. The user launches the application manager from an icon in the front panel. Users are notified when a new application is available on a server by additions (or deletions) to the list of icons in the application manager window. Programs and icons can be installed and pushed out to other workstations as an integral part of the installation process. The list of workstations that new software is installed on is configurable. The application manager comes preconfigured to include several utilities and programs.

The session manager

The session manager is responsible for the start up and shut down of a user session. In the CDE, applications that are made ***CDE aware*** are warned via an X Event when the X session is closing down. The application responds by returning a string that can be used by the session manager at the user's next login to restart the application. CDE can remember two sessions per user. One is the ***current*** session, where a snapshot of the currently running applications is saved. These applications can be automatically restarted at the user's next login. The other is the default login, which is analogous to starting an X session in the Motif window manager. The user can choose which of the two sessions to use at the next login.

Other CDE desktop tools

CDE 1.0 includes a set of applications that enable users to become productive immediately. Many of these are available directly from the front panel, others from the desktop or personal application managers. Common and productive desktop tools include:

Mail Tool
-- Used to compose, view, and manage electronic mail through a GUI. Allows the inclusion of attachments and communications with other applications through the

messaging system.

Calendar Manager
-- Used to manage, schedule, and view appointments, create calendars, and interact with the Mail Tool.

Editor
-- A text editor with common functionality including data transfer with other applications via the clipboard, drag and drop, and primary and quick transfer.

Terminal Emulator
-- An ***xterm*** terminal emulator.

Calculator
-- A standard calculator with scientific, financial, and logical modes.

Print Manager
-- A graphical print job manager for the scheduling and management of print jobs on any available printer.

Help System
-- A context-sensitive graphical help system based on Standard Generalized Markup Language (SGML).

Style Manager
-- A graphical interface that allows a user to interactively set their preferences, such as colors, backdrops, and fonts, for a session.

Icon Editor
-- This application is a fairly full featured graphical icon (pixmap) editor.

Application development tools

CDE includes two components for application development. The first is a shell command language interpreter that has built-in commands for most X Window system and CDE functions. The interpreter is based on ksh93 (The Korn Shell), and should provide anyone familiar with shell scripts the ability to develop X, Motif, and CDE applications.

To support interactive user interface development, developers can use the Motif Application Builder. This is a GUI front end for building Motif applications that generates C source code. The source code is then compiled and linked with the X and Motif libraries to produce the executable binary.

Application integration

CDE provides a number of tools to ease integration. The overall model of the CDE session is intended to allow a straightforward integration for virtually all types of applications. Motif and other X toolkit applications usually require little integration.

The task of integrating in-house and third party applications into a desktop, often the most difficult aspect of a desktop installation, is simplified by CDE. The power and advantage of CDE functionality can be realized in most cases without recompiling applications.

For example, Open Look applications can be integrated through the use of scripts that perform front-end execution of the application and scripts that perform pre- and post-session processing.

After the initial task of integrating applications so that they fit within session management, further integration can be done to increase their overall common ***look-and-feel*** with the rest of the desktop and to take advantage of the full range of CDE functionality. Tools that ease this aspect of integration include an ***Icon Editor*** used to create colour and monochrome icons. Images can be copied from the desktop into an icon, or they can be drawn freehand.

The ***Action Creation Utility*** is used to create action entries in the action database. Actions allow applications to be launched using desktop icons, and they ease administration by removing an application's specific details from the user interface.

The ***Application Gather*** and ***Application Integrate*** routines are used to control and format the application manager. They simplify installations so that applications can be accessible from virtually anywhere on the network.

Windows and the Window Manager

From a user's perspective, one of the first distinguishing features of Motif's ***look and feel*** is the ***window frame*** (Fig. 1.3). Every application window is contained inside such a frame. The following items appear in the window frame:

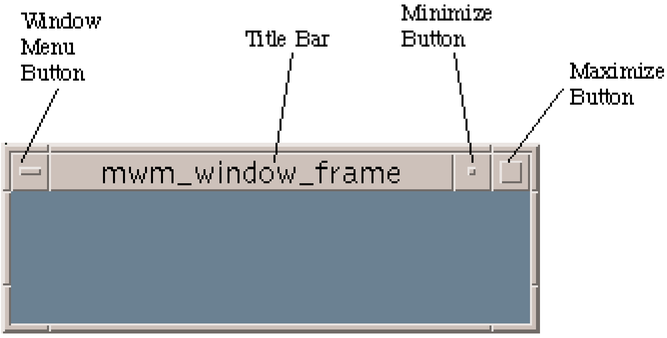


Fig. 1.3 The Motif Window Frame

Title Bar
: -- This identifies the window by a text string. The string is usually the name of the application program. However, an application's resource controls the label (Chapter).

Window Menu
: -- Every window under the control of ***mwm*** has a window menu. The application has a certain amount of control over items that can be placed in the menu. The ***Motif Style Guide*** insists that certain commands are always available in this menu and that they can be accessed from either mouse or keyboard selection. Keyboard selections are called ***mnemonics*** and allow routine actions (that may involve several mouse actions) to be called from the keyboard. The action from the keyboard usually involves pressing two keys at the same time: the **`Meta`** key and another key. The default window menu items and ***mnemonics*** are listed below and illustrated in Fig. 1.4:

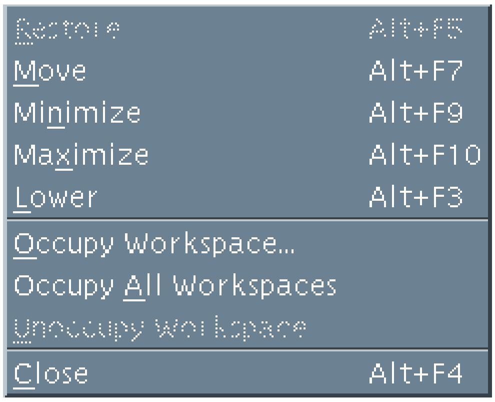


Fig. 1.4 The Window Menu

- **Restore (`Meta+F5`)** -- Restore window to previous size after iconification (***see*** below).
- **Move (`Meta+F7`)** -- Allows the window to be repositioned with a drag of the mouse.
- **Size (`Meta+F8`)** -- Allows the size of the window to be changed by dragging on the corners of the window.
- **Minimize (`Meta+F9`)** -- Iconify the window.
- **Maximize (`Meta+F10`)** -- Make the window the size of the root window, usually the whole of the display size.
- **Lower (`Meta+F3`)** -- Move the window to the bottom of the window stack. Windows may be ***tiled*** on top of each other (***see*** below). The front window being the top of the stack.
- **Close (`Meta+F4`)** -- Quit the program. Some simple applications (Chapter) provide no ***internal*** means of termination. The `Close` option being the only means to achieve this.

Minimize Button
-- another way to iconify a window .

Maximize Button
-- another way to make a window the size of the root window .

The window manager must also be able to manage multiple windows from multiple client applications. There are a few important issues that need to be resolved. When running several applications together, several windows may be displayed on the screen. As a result, the display may appear cluttered and hard to navigate. The window manager provides two mechanisms to help deal with such problems:

Active Window
-- Only one window can receive input at any time. If you are selecting a graphical object with a mouse, then it is relatively easy for the window manager to detect this and schedule appropriate actions related to the chosen object. It is not so easy when you enter data or make selections directly from the keyboard. To resolve this only one window at a time is allowed ***keyboard focus***. This window is called the ***active window***. The selection of the active window will depend on the system configuration which the user typically has control over. There are two common methods for selecting the active window:

Focus follows pointer
-- The active window is the window is the window underneath mouse pointer.

Click-to-type
-- The active window is selected, by clicking on an area of the window, and remains active until another window is selected no matter where the mouse

points.

When a window is made active its appearance will change slightly:

- Its outline frame will become shaded.
- The cursor will change appearance when placed in the window.
- The window may jump, or be ***raised*** to the top of the window stack.

The exact appearance of the above may vary from system to system and may be controlled by the user by setting environment settings in the window manager.

Window tiling

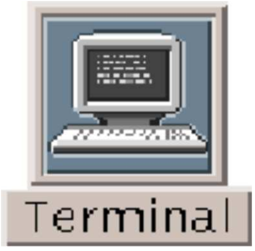
-- Windows may be stacked on top of each other. The window manager tries to maintain a three-dimensional ***look and feel*** . Apart from the fact that buttons, dialog boxes appear to be elevated from the screen, windows are shaded and framed in a three-dimensional fashion. The top window (or currently active window) will have slightly different appearance for instance.

The window menu has a few options for controlling the tiling of a window. Also a window can be brought to the top of the stack, or ***raised*** by clicking a part of its frame.

Iconification

-- If a window is currently active and not required for input or displaying output then it may be ***iconified*** or ***minimised*** thus reducing the screen clutter. An icon (Fig. 1.5) is a small graphical symbol that represents the window (or application). It occupies a significantly less amount of screen area. Icons are usually arranged around the perimeter (typically bottom or left side) of the screen. The application will still be running and occupying computer memory. The window related to the icon may be reverted to by either double clicking on the icon, or selecting ***Restore*** or ***Maximise*** from the icon's window menu.

Figure 1.5: Sample Icon from Xterm Application



The Root Menu

The ***Root Menu*** is the main menu of the window manager. The root menu typically is used to control the whole display, for example starting up new windows and quitting the desktop. To display the Root menu:

- Move the mouse pointer to the Root Window.
- Hold down the left mouse button.

The default Root Menu has the following The root menu can be customised to start up common applications for example. The root menu for the ***mwm*** (Fig. 1.6) and ***dtwm*** (Fig. 1.7) have slightly different appearance but have broadly similar actions, which are summarised below:

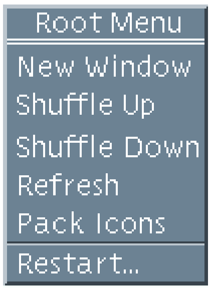


Fig. 1.6 The *mwm* Root Menu

Fig. 1.7 The CDE *dtwm* Root Menu

Program
(***dtwm***) -- A sub-menu is displayed that allows a variety of programs to be called from the desktop, for example to create a new window. The list of available programs can be customised from the desktop.

New Window
(***mwm***) -- Create a new window which is usually an ***Xterm*** window.

Shuffle Up
-- Move the bottom of the window stack to the top.

Shuffle Down
-- Move the top of the window stack to the bottom.

Refresh
-- Refresh the current screen display.

Restart
-- Restart the Workspace.

Logout
(***dtwm***) -- Quit the Window Manager.

Exercises

Exercise 12158

Exercise~\ref{ex.cde1}

Add an application to the application manager

Exercise 12159

Practice opening, closing and moving windows around the screen and to/from the background/foreground. Get used to using the mouse and its buttons for such tasks.

Exercise 12160

Figure out the function of each of the three mouse buttons. Pay particular attention to the different functions the buttons in different windows (applications) and also when the mouse is pointing to the background.

Exercise 12161

Find out how to resize windows etc. and practice this.

Exercise 12162

Fire up the texteditor of your choice (You may use `dtpad` (basic but functional), `textedit` application (SOLARIS basic editor), emacs/Xemacs, or vi) and practice editing text files. Create any files you wish for now. Figure out basic options like cut and paste of text around the file, saving and loading files, searching for strings in the text and replacing strings.

Particularly pay attention in getting used to using the Key Strokes and / or mouse to perform the above tasks.

Exercise 12163

Use Unix Commands to

1. Copy a file (created by text editor or other means) to another file called spare.
2. Rename your original file to b called new.
3. Delete the file spare.
4. Display your original file on the terminal.
5. Print your file out.

Exercise 12164

Familiarise yourself with other UNIX functions by creating various files of text etc. and trying out the various functions listed in handouts.

Dave Marshall
1/5/1999