

Operating Systems and the User Interface

Operating System

When you turn on your computer, there's a lot going on inside, and the real man behind the curtain handling the necessary tasks is the operating system.

Most desktop or laptop PCs come pre-loaded with Microsoft Windows. Macintosh computers come pre-loaded with Mac OS X. Many corporate servers use the Linux or UNIX operating systems. The operating system (OS) is the first thing loaded onto the computer; without the operating system, a computer is useless.

More recently, operating systems have started to pop up in smaller computers as well. The computers used in these little **devices** have gotten so powerful that they can now actually run an operating system and **applications**.

The purpose of an operating system is to organize and control **hardware** and **software** so that the device it lives in behaves in a flexible but predictable way.

Not all computers have operating systems. The computer that controls the microwave oven in your kitchen, for example, doesn't need an operating system. It has one set of tasks to perform, very straightforward input to expect (a numbered keypad and a few pre-set buttons) and simple, never-changing hardware to control. For a computer like this, an operating system would be unnecessary. Instead, the computer in a microwave oven simply runs a single **hard-wired** program all the time.

For other devices, an operating system creates the ability to:

- serve a variety of purposes
- interact with users in more complicated ways
- keep up with needs that change over time

All desktop computers have operating systems. The most common are the Windows family of operating systems developed by Microsoft, the Macintosh operating systems developed by Apple and the UNIX family of operating systems (which have been developed by a whole history of individuals, corporations and collaborators). There are hundreds of other operating systems available for

special-purpose applications, including robotics, manufacturing, real-time control systems and so on.

In any device that has an operating system, there's usually a way to make changes to how the device works. One of the reasons operating systems are made out of **portable code** rather than permanent physical circuits is so that they can be changed or modified without having to redesign the whole device.

For a desktop computer user, this means you can add a new security update, system patch, new application or even an entirely new operating system rather than junk your computer and start again with a new one when you need to make a change. As long as you understand how an operating system works and how to get at it, in many cases you can change some of the ways it behaves.

User Interface

A **user interface** (UI) brings structure to the interaction between a user and the computer. In the last decade, almost all **development** in user interfaces has been in the area of the **graphical user interface** (GUI), with two models, Apple's Macintosh and Microsoft's Windows, receiving most of the attention and gaining most of the market share. Linux operating system also supports a graphical user interface.



Screen shot copyright © 2003 Red Hat, Inc. All rights reserved.
Reused with permission from [Red Hat, Inc.](http://www.redhat.com)

Screen shot of Red Hat's Linux operating system

The user interface is a program or set of programs that sits as a layer above the operating system itself. The core operating-system functions (the management of the computer system) lie in the **kernel** of the operating system. The User Interface is separate, though it may be tied tightly to the **kernel** beneath.

There are other user interfaces, some graphical and some not, for other operating systems.

Unix, for example, has user interfaces called **shells** that present a user interface more flexible and powerful than the standard operating system **text-based interface**. Programs such as the Korn Shell and the C Shell are text-based interfaces that add important utilities, but their main purpose is to make it easier for the user to manipulate the functions of the operating system. There are also **graphical user interfaces**, such as X-Windows and Gnome, that make Unix and Linux more like Windows and Macintosh computers from the user's point of view.

When IBM released the PC, it came with an operating system called DOS. Like just about every operating system at the time, DOS had a command-line interface. You typed in commands like DIR or COPY, and the operating system would respond. The advantage was that these systems were simple to program and they fit well with the character-based screens that were common at the time. But "normal people" (meaning, non-geeks) had a lot of trouble feeling comfortable with DOS.

Then in 1984 there was an event that changed everything. Apple released the Macintosh computer with its unbelievable **Graphical User Interface (GUI)**. Because we all use GUIs every day, it is hard for us to understand today how revolutionary the Mac was. But if you ask people who lived through the transition, many of them can actually remember the day they saw their first Mac.

It was so amazing, in fact, that everyone uses a GUI today. We would be lost without the graphical user interface. The thought of trying to navigate the Web from a command line is too painful to even imagine.





