

Operating Systems: Three Easy Steps

Kunwar Shaanjeet Singh Grover

Contents

1	Introduction	3
1.1	Virtualisation	3
1.1.1	CPU Virtualisation	3
1.1.2	Memory Virtualisation	3
1.2	Concurrency	3
1.3	Persistence	3

Chapter 1

Introduction

1.1 Virtualisation

The OS takes a **physical** resources (such as the processor , or memory, or a disk) and transforms it into a more general, powerful, and easy-to-use **virtual** for of itself. Thus, we sometimes refer to the operating system as a **virtual machine**. This general technique of transforming is called **virtualisation**.

1.1.1 CPU Virtualisation

Turning a single CPU into a seemingly infinite number of CPUs and thus allowing many programs to seemingly run at once is what we call **virtualizing the CPU**.

1.1.2 Memory Virtualisation

Each process accesses its own **virtual address spaces**, which the OS somehow maps onto the physical memory of the machine. Exactly how this is accomplished is what we study.

1.2 Concurrency

Concurrency is a conceptual term to refer to a host of problems that arise, and must be addressed when working on many things at once (i.e. concurrently) in the same program.

1.3 Persistence

The software in the operating system that usually manages the disk is called the **file system**; it is this responsible for storing any files the user creates in a

reliable and efficient manner on the disks of the system. The file system is the part of OS in charge of managing persistent data. What techniques are needed to do so? What mechanisms and policies are required to do so with a high probability?