Introduction

CS 537: Introduction to Operating Systems

Louis Oliphant

University of Wisconsin - Madison

Fall 2024

What Is An Operating System?

An ***operating system (OS)*** *is system software that* ***manages computer hardware and software resources****, and* ***provides common services*** *for computer programs. – wikipedia*

OS Role #1 – Common Services

- Abstraction – Provide a standard library for resources
- What is a resource?
  Anything valuable (e.g., CPU, Memory, disk)
- What abstraction does OS provide for each resource?
  - CPU: processes and threads
  - Memory: address space
  - Disk: files and directories

Why OS Abstracts Resources?

- Applications **reuse** common facilities
- Make different devices **look the same**
- Provide **higher-level or more useful functionality**

Challenges

- What are the correct abstractions?
- How much hardware should be exposed?

OS Role #2 – Resource Manager

- Protect Applications from one another
- Provide efficient access to resources (cost, time, energy)
- Provide fair access to resources

Challenges

- What are the correct mechanisms?
- What are the correct policies?

OS Goals

An OS manages resources and provides access to them through a set of common services in order to make those resources:

- Easy to use
- Reliable to access
- Fair access
- Protected from improper access
- Maximize Performance (i.e. little OS overhead)

Course Organization – Three Easy Pieces

Three Conceptual Pieces:

1. Virtualization
2. Concurrency
3. Persistence

Virtualization

Make each application believe it has **each resource to itself**

Demo – Virtualize CPU and memory

Concurrency

Events occur simultaneously and may interact with one another

- Easier case

  Hide concurrency from **independent** processes.
- Trickier case
  - Manage concurrency with **interacting** processes.
  - OS Provides abstractions (locks, semaphores, condition variables, shared memory, critical sections) to processes
  - Need to ensure processes do not deadlock

Demo – Interacting threads must coordinate access to shared data

Persistence

- Lifetime of data is longer than lifetime of any process
- Machine may be rebooted, lose power or crash unexpectedly

OS:

- Provides abstractions: **file**, **directory**, **link**
- Handles **correctness** with unexpected failures
- Ensures **performance**: disks are slow, many optimizations needed!

Demo – writing to file system

Advanced Topics

- Network File Systems
- Solid State Devices
- Cloud and Distributed Computing

Why Study OS?

- Build, modify, or administer an OS
- Understand system performance
  - Behavior of OS impacts entire machine
  - Tune workload performance
  - Apply knowledge across many layers
- Fun and challenging to understand large, complex systems

Next Steps

- Check out course web page
- Read project 1 instructions
  Details in discussion section

WELCOME TO CS 537!