## Introduction

2566

## Computer Performance Over Time

|                                | 1981                | 1997                   | 2014                  | Factor<br>(2014/1981) |
|--------------------------------|---------------------|------------------------|-----------------------|-----------------------|
| Uniprocessor speed (MIPS)      | 1                   | 200                    | 2500                  | 2.5K                  |
| CPUs per computer              | 1                   | 1                      | 10+                   | 10+                   |
| Processor MIPS/\$              | \$100K              | \$25                   | \$0.20                | 500K                  |
| DRAM Capacity (MiB)/\$         | 0.002               | 2                      | 1K                    | 500K                  |
| Disk Capacity (GiB)/\$         | 0.003               | 7                      | 25K                   | 10M                   |
| Home Internet                  | 300 bps             | 256 Kbps               | 20 Mbps               | 100K                  |
| Machine room network           | 10 Mbps<br>(shared) | 100 Mbps<br>(switched) | 10 Gbps<br>(switched) | 1000                  |
| Ratio of users<br>to computers | 100:1               | 1:1                    | 1:several             | 100+                  |

From Thomas Anderson and Michael Dahlin, Operating Systems Principles & Practice Volume I, 2<sup>nd</sup> edition, Recursive Books, 2015

# Early Operating Systems: Computers Very Expensive

- One application at a time
  - Had complete control of hardware
  - OS was runtime library
  - Users would stand in line to use the computer
- Batch systems
  - Keep CPU busy by having a queue of jobs
  - OS would load next job while current one runs
  - Users would submit jobs, and wait, and wait, and

# Time-Sharing Operating Systems: Computers and People Expensive

- Multiple users on computer at same time CPU 1ตัว สามาถเข้าใช้งานได้พร้อมกัน
  - Multiprogramming: run multiple programs at same time รันหลายๆโปรแกรมได้พร้อมกัน
  - Interactive performance: try to complete everyone's tasks quickly ทำให้คอม ตอบสนองต่อผู้ใช้
  - As computers became cheaper, more important to optimize for user time, not computer time พอจากคอมแพงค่าจ้างถูก พนักงานอยู่ว่างๆ 3 วัน แต่ตอนนี้ราคาถูกลง คนค่าาจ้างสูงขึ้น เราจะทำยังไงใหดอบสนองต่อผู้ใช้ได้

ปัจจุบันคอมพิวเตอร์ราคาถูกลง ค่าจ้างแพงขึ้น

## Today's Operating Systems:

Computers Cheap ยุคปัจจุบันคอมพิวเตอร์มีราคาถูกมากเมื่อเปรียบเทียบกับสมัยเมื่อก่อน การทำงานก็จะอยู่ในหลายๆส่วนตามที่ list ด้านล่าง ก็ล้วนมีชิ้นส่วนคอมพิวเตอร์ทั้งนั้น จะเห็นได้ว่าการจัดการทั้งหลาย จะเริ่มต้นเป็นรูปเป็นร่างตอนยุคสมัยนี้

- Smartphones
- Embedded systems
- Laptops
- Tablets
- Virtual machines
- Data center servers

1975 - ปัจจุบัน

## Tomorrow's Operating Systems

- Giant-scale data centers
- Increasing numbers of processors per computer
- Increasing numbers of computers per user
- Very large scale storage

กิจกรรม #1

•ในความคิดเห็นของนักศึกษา ระบบปฏิบัติการมี บทบาทหน้าที่หลักอะไรบ้าง

## Roles of the Operating System

#### • Referee:

- Resource allocation among users, applications
- Isolation of different users, applications from each other
- Communication between users, applications

#### • Illusionist

- Each application appears to have the entire machine to itself
- Infinite number of processors, (near) infinite amount of memory, reliable storage, reliable network transport

#### Glue

• Libraries, user interface widgets, ... เชื่อมประสานชิ้นส่วนต่างๆเข้าด้วยกันในระบบ OS ก็คือ เชื่อม User เข้ากับ ระบบ

กิจกรรม #2

•ระบบปฏิบัติการคืออะไร?

## What is an Operating System?

- A set of software that manage computer's resources for its users and their applications
  - May visible or invisible to the user
  - 2 major kinds
    - General purpose OS
    - Specific purpose OS

บริหารจัดการ ทรัพยากรที่เครื่องคอมนั้นๆมีอยู่ จัดสรรให้กับ USER ใช้งาน Application ทำงานให้มีประสิทธิภาพมากที่สุด

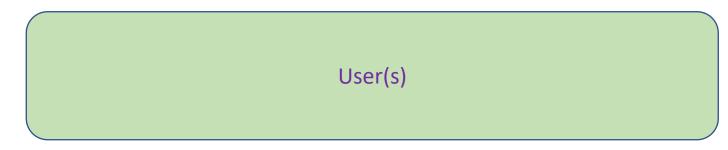
•หากนักศึกษาต้องประเมินระบบปฏิบัติการหนึ่ง นักศึกษาจะประเมินด้านใดบ้าง และแต่ละด้านจะ วัดอย่างไร

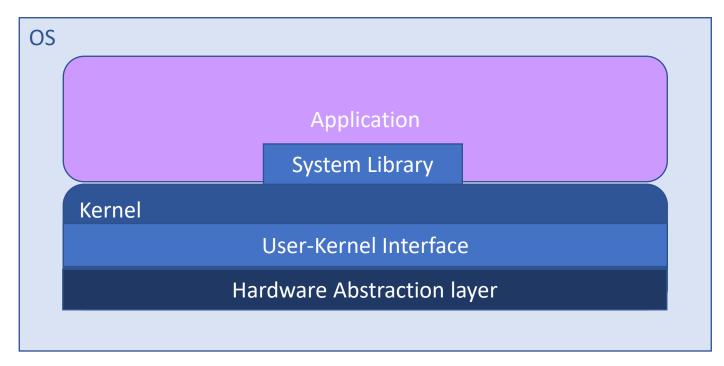
### Operating System Evaluation

- Reliability and Availability
- Security
- Portability
  - AVM, API, HAL
- Performance
  - Overhead, efficiency
  - Fairness, response time, throughput
  - Performance predictability
- Adoption

## Design Tradeoffs

- Must balance between the 5s
- Examples
  - Preserves legacy API  $\rightarrow$  Portability  $\spadesuit$ , reliable  $\Psi$ , secure  $\Psi$
  - ullet Breaking an abstraction ullet Performance ullet, Portability ullet, Reliability ullet





Hardware