PENNSTATE

Intro to Operating Systems

CMPSC 473

PENNSTATE

Administrative Details

Instructor

Trent Jaeger (trj1@psu.edu)
W359 Westgate

Office Hours: Tu: 3:30-4:45PM in office
Th: 9:30-10:45AM in office
or by appt.

Zoom info on Canvas – in case we need it

PENNSTATE

About me

- **Trent Jaeger**
  - Research focus: Computer Security
    - Operating Systems and Software Security
  - IBM Research for nine years
  - Penn State for 18 years

Welcome!

TAs

- Aditya Basu (aditya.basu@psu.edu) : Tu,W - 10:00am-11:15am
- Frank Capobianco (fnc110@psu.edu) : W,Fr - 2:30pm-3:45pm
- Ankush Mishra (aam6386@psu.edu) : M,Fr - 10:30am-11:45am
- David Reinoso (dar5654@psu.edu) : W,Th - 4:00pm-5:15pm
- Niramay Vaidya (nvv5143@psu.edu) : M,W - 9:00-10:15am

In Bldg 300 West College Ave

Please go over the Canvas Discussions page to see if your queries have been already answered. If not, and if you feel that your question may be useful to a larger audience, you can use the Discussions page to post your questions. **Please do NOT post pieces of code here**, which will constitute a violation of Academic Integrity.

PENNSTATE

Course Description

- Concepts, algorithms, and implementation of operating systems
- Important to understand the concepts and also how they are implemented
- Course Text --- "Operating Systems : Three Easy Pieces" by R. Arpaci-Dusseau and A. Arpaci-Dusseau
  - http://pages.cs.wisc.edu/~remzi/OSTEP/

PENNSTATE

Prerequisites

- **311 and 331 are pre-requisites.**
- **Knowledge of C, and basic UNIX shell interface.**
- **Knowledge of how high-level programs execute on the hardware**
  - **how does a high-level program get translated to assembly/binary, linker, loader, runtime (stack-based), program layout – code, data, stack, heap, etc.**
- **Basic knowledge of hardware**
  - **processors, instruction sets, caches and memory hierarchy, peripherals, DMA, disks, etc.**

If you are not sure about any of this, you either need to take the corresponding prerequisite courses, or read them by yourselves. Note that you have been pre-warned, and it will be your problem if you cannot keep up with all this material.

Projects

- 4 programming projects (Deadlines: Jan 27, Feb 21, Mar 30, Apr 25)
- Each project will require several weeks of work – so no excuses!
- Knowledge of C language, toolchain, and debugging
- Project 2 will be in teams of 2 people. Others will be individual. Can chose project members across sections.
  - Will have minimal requirements for team members to contribute to P2
- Send email to TAs with team members, they will track the teams, and can team you up if needed.
- Start early on the projects. Note that there will be no extensions.

- Department's AI policy regarding coding projects:
  - https://www.eecs.psu.edu/students/resources/EECS-CSE-Academic-Integrity.aspx

- Cheating Penalty: 0 on project and additional penalty equal to the credit (points) for the project (like losing two projects)

Exams

- Midterm during class hours (Feb 28) + Final during finals week
- You have until Jan 27 to email me regarding any conflicts you may have for exams, together with a very valid reason.
- After this date, no requests for conflicts will be entertained!
- All exams are closed book/notes and “non comprehensive”.
- Some coding questions are expected

Quizzes

- In class – middle of lecture
- 1 or 2 problems – complete in 10 minutes
- Likely all will be announced
- On Canvas

Schedule/Grading

	Percent of Grade	Date
Project 0	2 %	Due Friday, Jan 27 (by 11.59 PM)
Project 1	9 %	due Tuesday, Feb 21 (by 11.59 PM)
Project 2	9 %	due Thursday, Mar 30 (by 11.59 PM)
Project 3	10 %	due Tuesday, Apr 25 (by 11.59 PM)
Midterm	25 %	Tuesday, Feb 28 (in class)
Final	35 %	during Finals Week
Quizzes	10 %	On Canvas

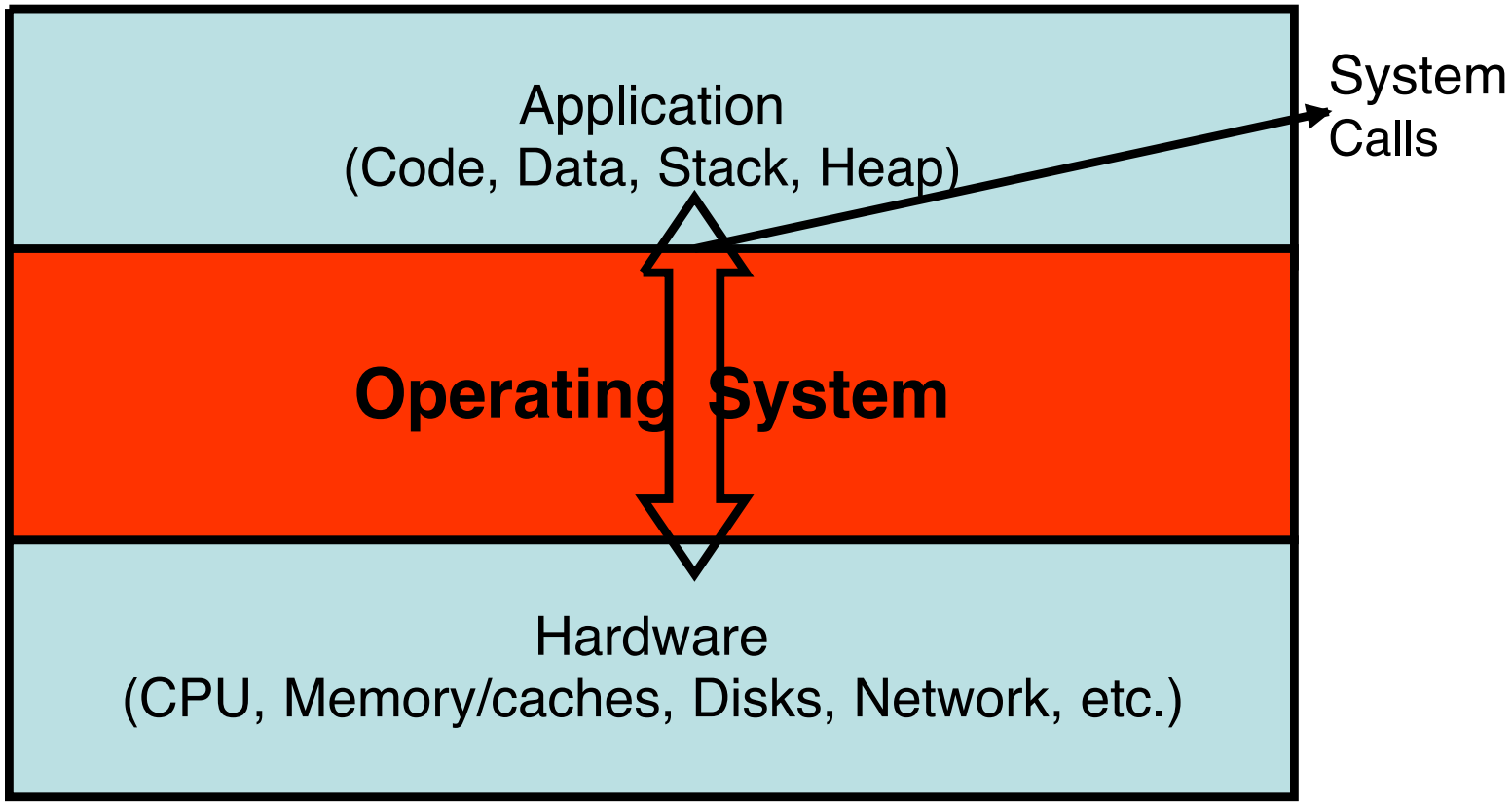
Class attendance/participation

- **In person class/exams**
- **Though there is a course text, the instructor may present additional material.**
- **You are responsible for studying and understanding everything covered in class (not just what is in the text or the slides) for the exams.**
- **Active class participation will ensure you closely follow what is going on.**
- **Please ask questions in class to clarify any doubts you may have.**

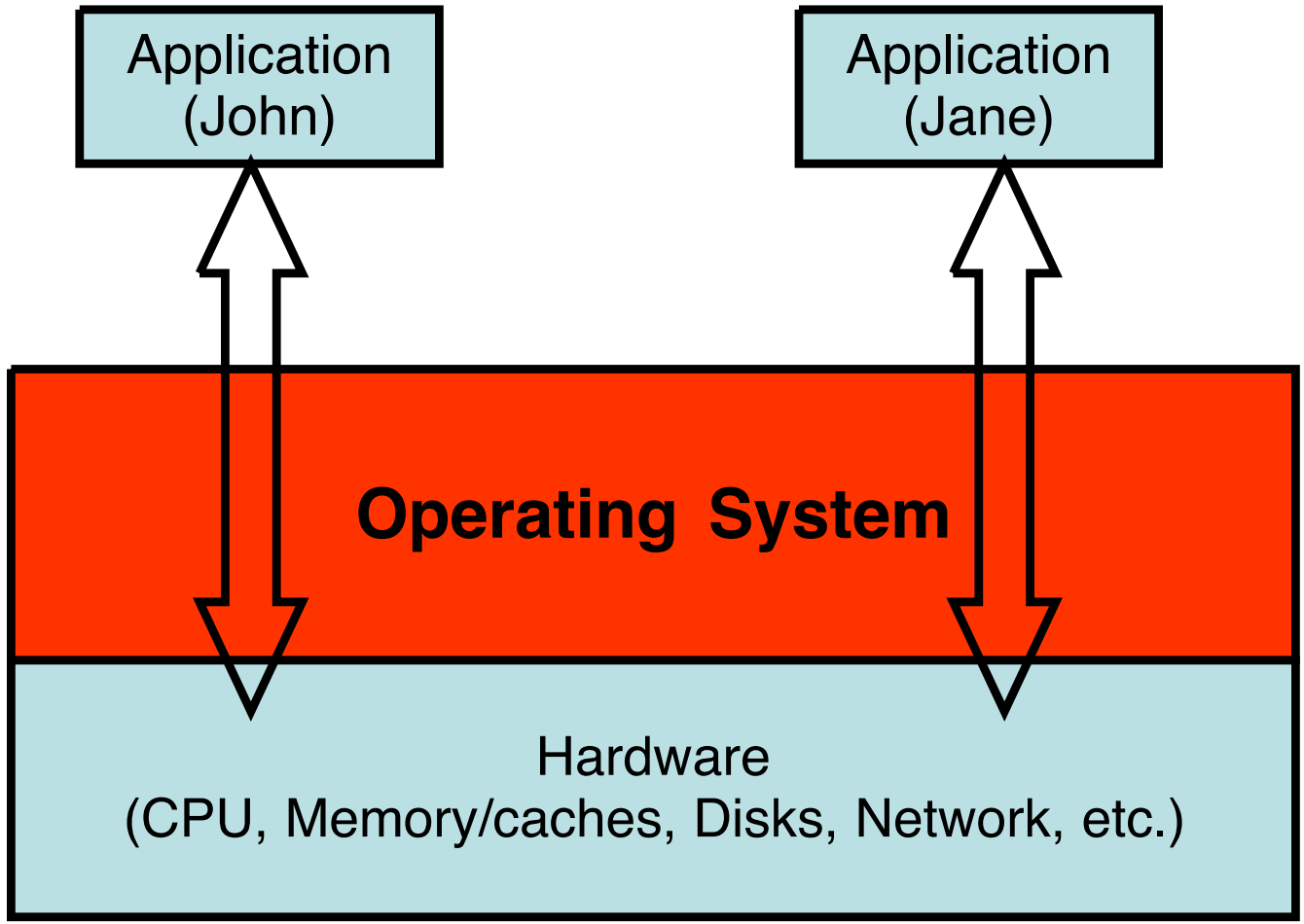
PENNSTATE

OS Overview

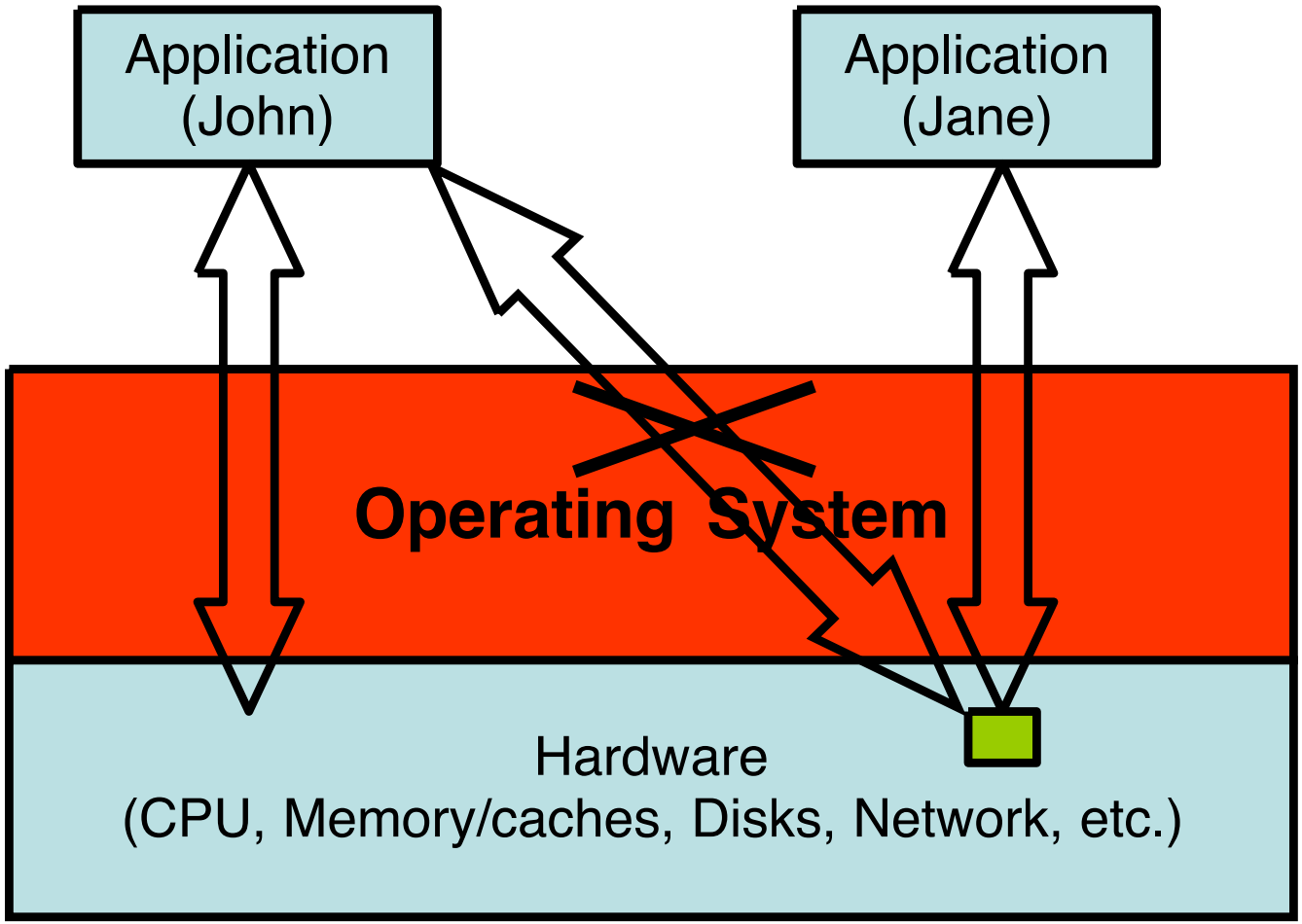
What is an Operating System (OS)?



Role: Service Provider (abstract/convenient interface to hardware)



Role: Resource manager/multiplexer (to avoid applications stepping on each other's toes)



Role: Resource Protection enforcer

PENNSTATE

OS Roles (summary)

- Service provider (abstract h/w interface)
- Resource manager/multiplexer
- Protection enforcer

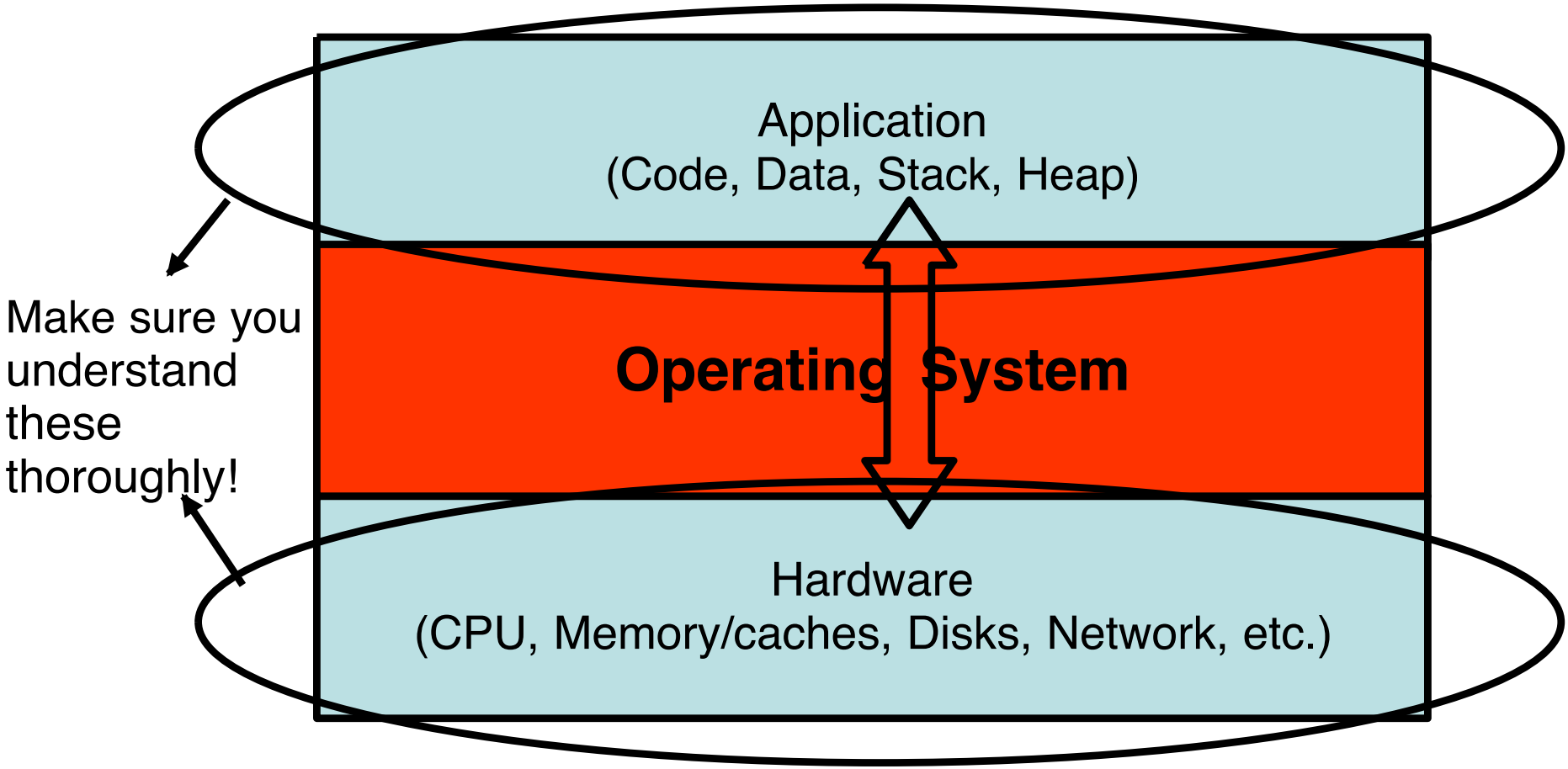
PENNSTATE

OS Roles (More Detail)

- **HW resources we will examine:**
  - **CPU: Abstracted to Processes/Threads**
  - **Memory: Abstracted to Regions/Pages**
  - **Disks: Abstracted to Files/Directories**
  - **Other I/O devices: Device drivers**
- **Access efficiently, atomically, securely:**
  - **Schedulers**
  - **Concurrency Primitives**
  - **Access Control and Authentication**

PENNSTATE

Before understanding operating systems ...



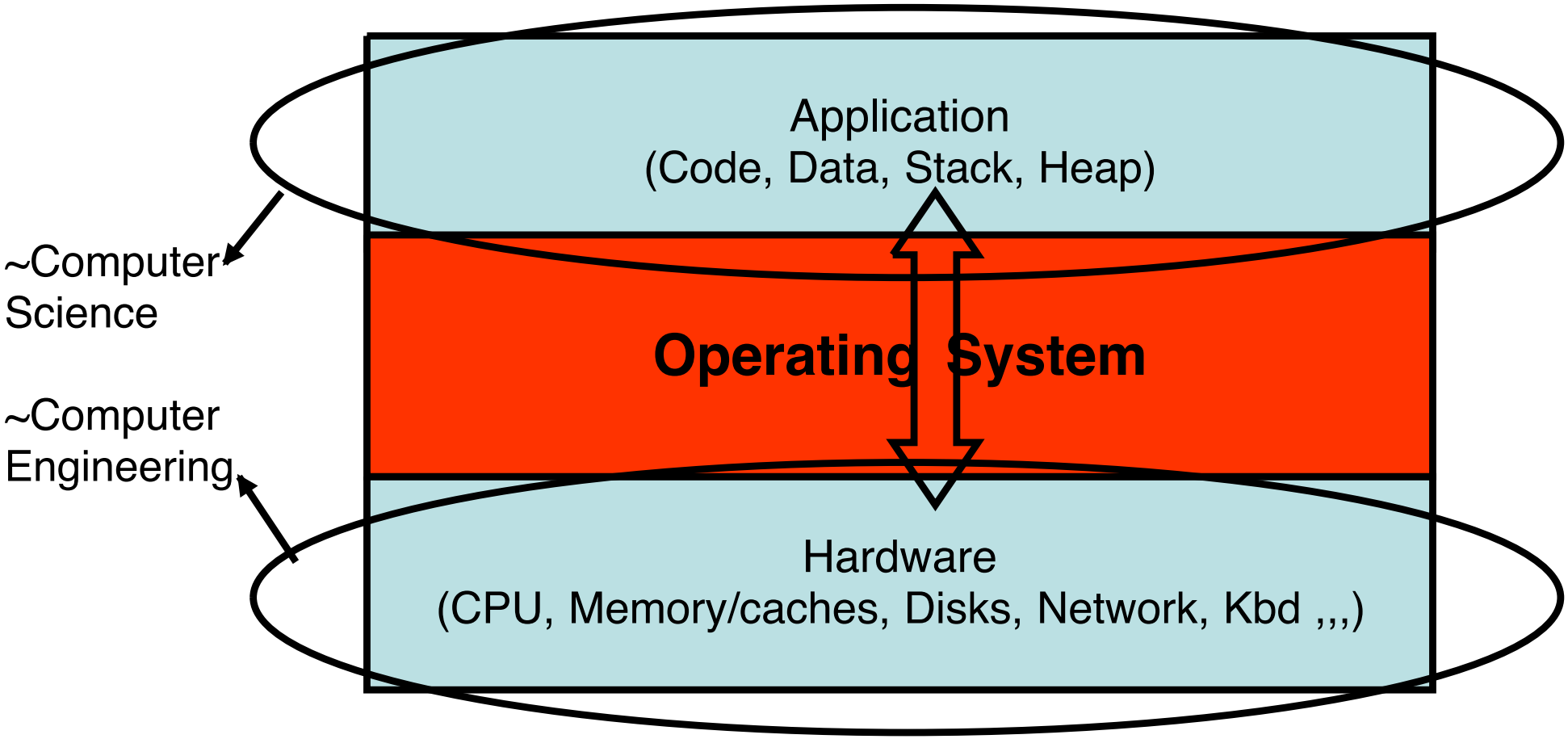
These are pre-requisites to this course

PENNSTATE

Why Should You Learn OS?

PENNSTATE

From Your Perspectives



PENNSTATE

OS Impact

- **Perhaps most of you will not become "kernel hackers"**
- **But, the way operating systems work impacts how your programs run and whether your HW is effective**

PENNSTATE

User Programs

- **Program operation is determined by the operating system**
- **Performance – Why does my program run slowly?**
- **Reliability – Why is my program running incorrectly?**
- **Understanding how operating systems work can help solve these kinds of problems – in many forms**

PENNSTATE

Program Example

- We were trying to write a program to assess a side channel vulnerability
  - Disclaimer: Seek approval before writing attack programs
- Side channel: X server processes unicode characters (e.g., the Euro symbol, €) much more slowly than ascii characters
- So, if your password has an unicode character, we could detect that by measuring the time between characters
- Password recovery: Could an adversary use this knowledge to recover someone's password illicitly?

PENNSTATE

Program Example

- We wrote a program to record the time before and after each character is entered (for X server)
- Then measure the delay for X server processing (or Unicode char we insert)
- Requirement: Our program must run right before and right after X server to measure the delay accurately
- Problem: Many processes run on a computer, so our program was not run regularly when we required
- Measurements were very noisy
- What determines when a program (process) is run?

PENNSTATE

Linux Scheduler

- **Problem**: Many processes that run on a computer, so our program was not run regularly when we required (i.e., right before and after X server)

- **Solution**: By learning more about how the Linux scheduler worked, we were able to redesign our program to ensure that it ran when required
- We will learn about how processes are scheduled here

PENNSTATE

Hardware Features

- **Hardware designers introduce new features, but whether these features are effective depends on how OSes utilize them**
- **Performance – Provide efficient use of feature?**
- **Reliability – Does the feature work as intended?**
- **Understanding how operating systems work can help solve these kinds of problems – in many forms**

PENNSTATE

HW Example

- **Intel introduced the Software Guard Extensions (SGX) to run software in a protected (encrypted) environment on your computer**
- **Intent: SGX memory is encrypted, so no attacker can see the secret data your program uses/generates**
- **Does this hardware feature work as intended?**

PENNSTATE

HW Example

- **Does this hardware feature work as intended?**
- **Problem: The OS controls access to other hardware used by an SGX program**

- **End result is that the OS can control the execution of an SGX program (e.g., run one instruction at a time) to gather information to infer secret data (e.g., see which execution paths are run)**
- **We will learn about how an OS helps programs run**

PENNSTATE

Questions?