

Overview of System Systems Programming

(CST-210)

Dr. A. P. Mazumdar

Outline

- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

Revisit C Compilation

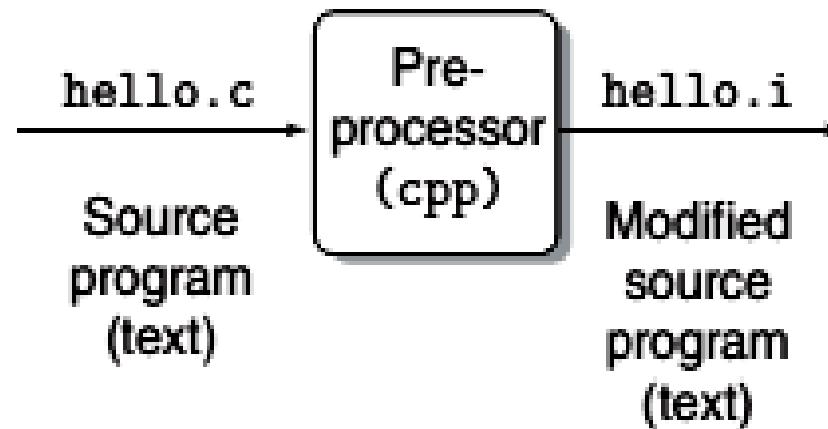
- ▶ A very simple C program:

```
#include<stdio.h>
void main() {
    printf("Hello World \n");
}
```

- ▶ We stored the program in *hello.c*

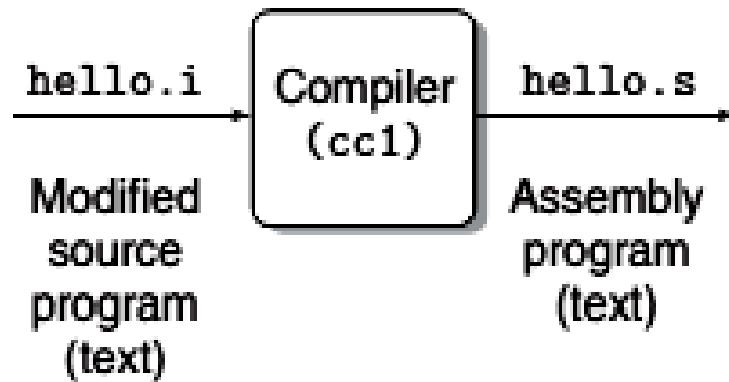
Revisit C Compilation (contd.)

▶ STEP 1:



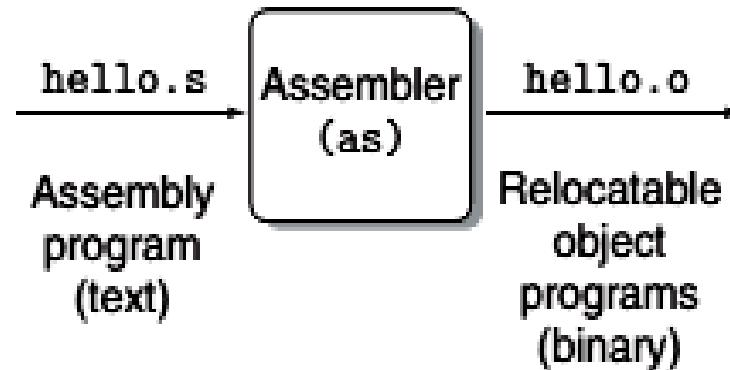
Revisit C Compilation (contd.)

▶ STEP 2:



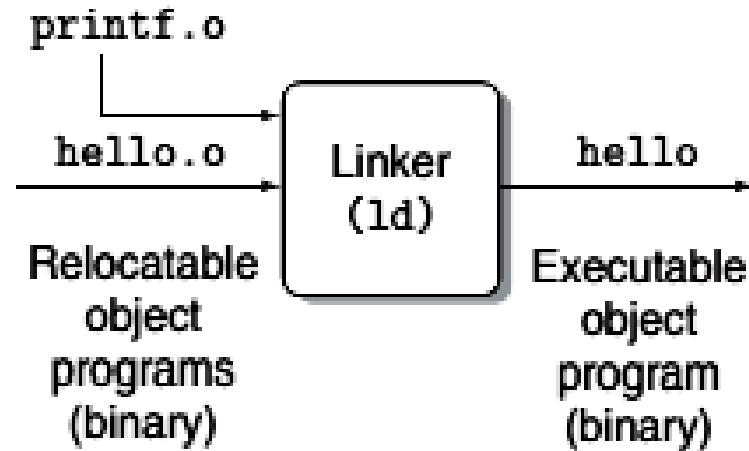
Revisit C Compilation (contd.)

▶ STEP 3



Revisit C Compilation (contd.)

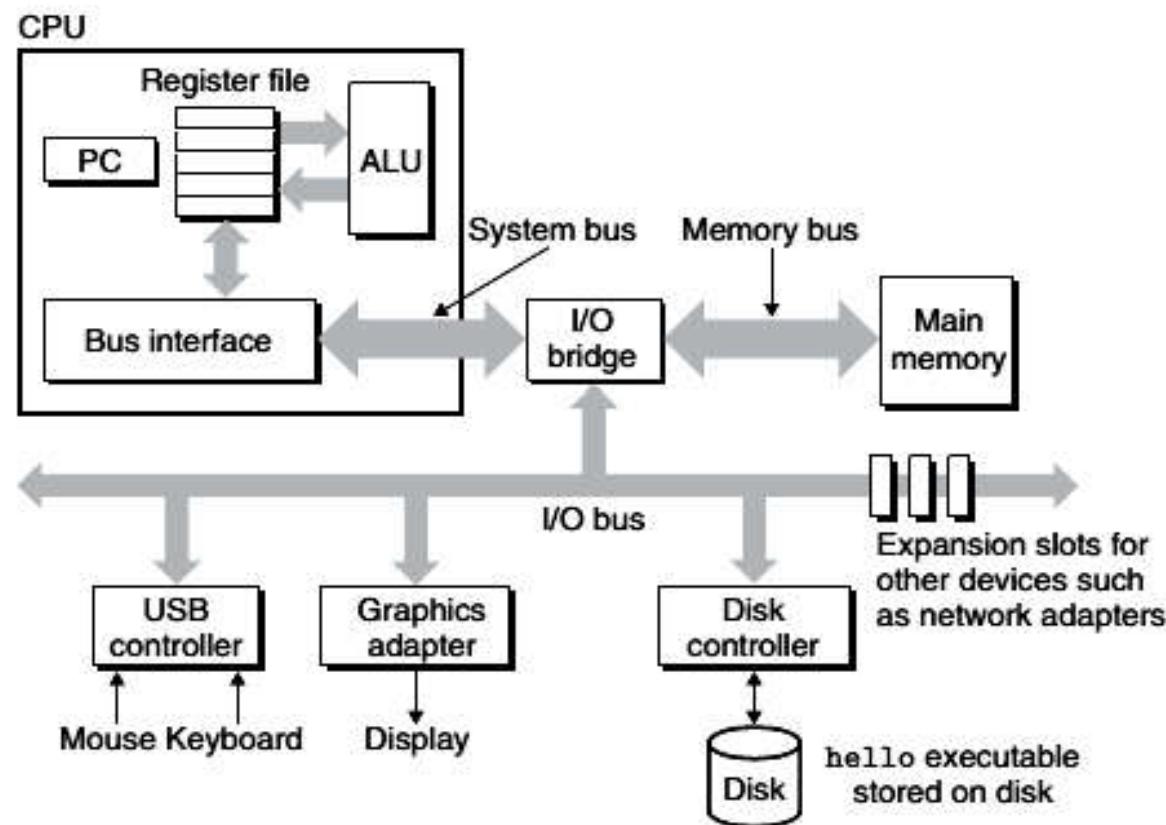
▶ STEP 4:



Outline

- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

Tour of a Computer System



Outline

- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

Running a C program

- ▶ Compile:

```
$ gcc -o hello hello.c
```

- ▶ Run

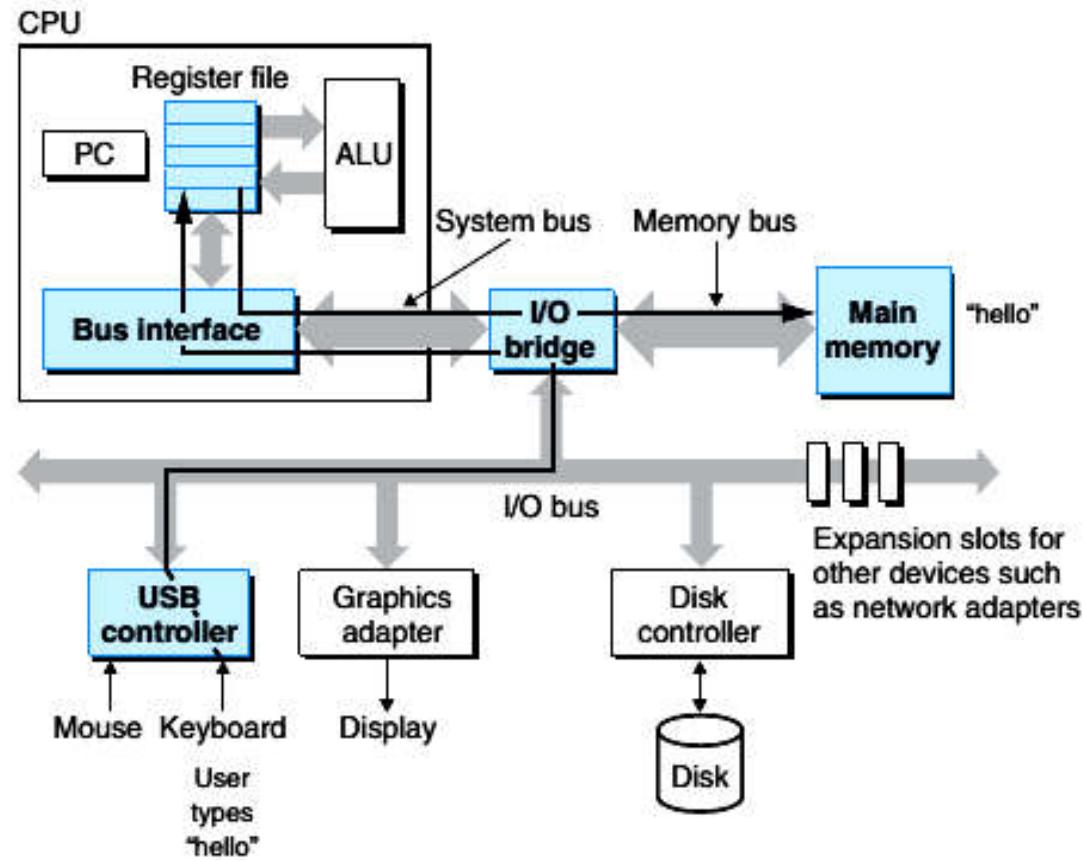
```
$ ./hello
```

```
Hello World
```

```
$ _
```

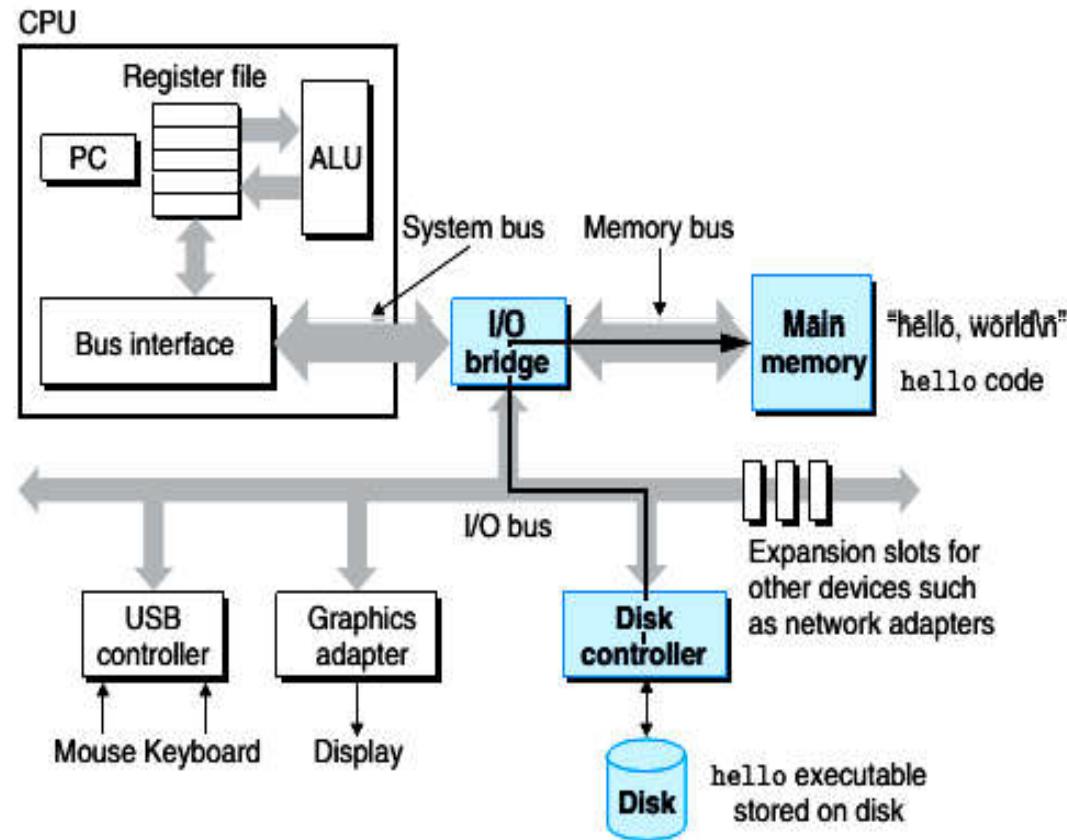
Running a C program (contd.)

- ▶ Reading ./hello



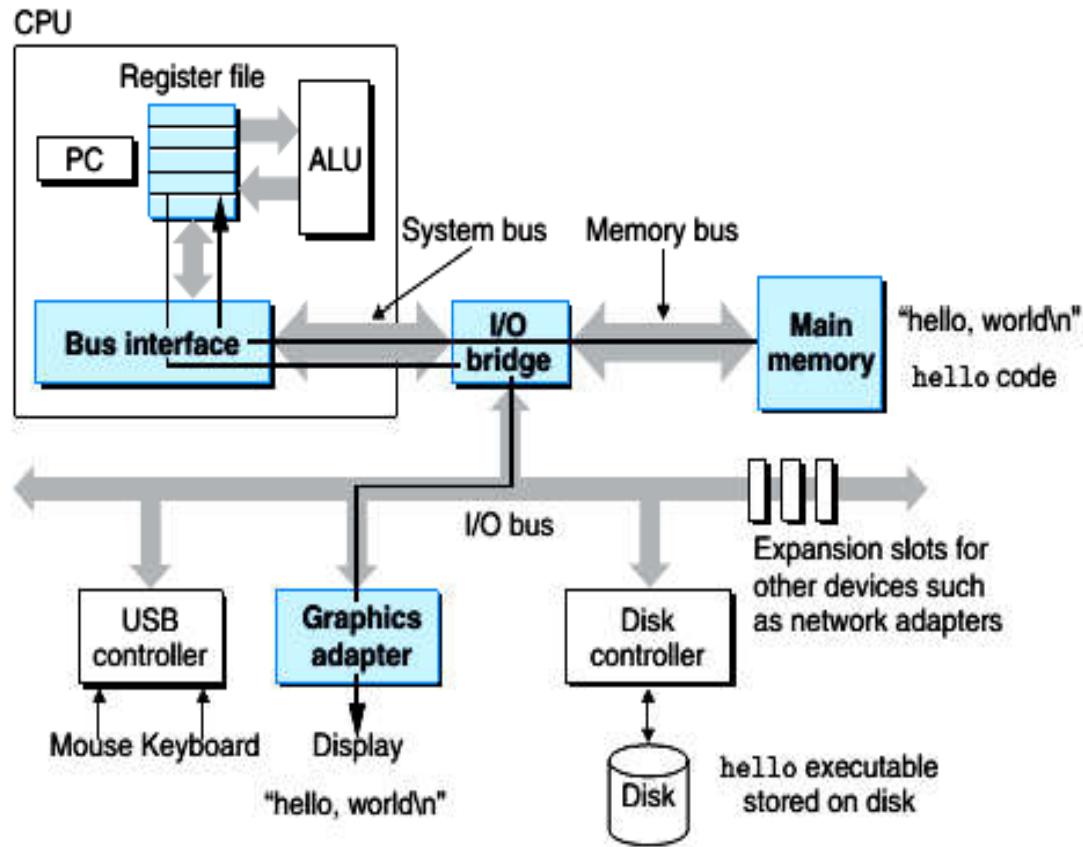
Running a C program (contd.)

▶ Loading the executable



Running a C program (contd.)

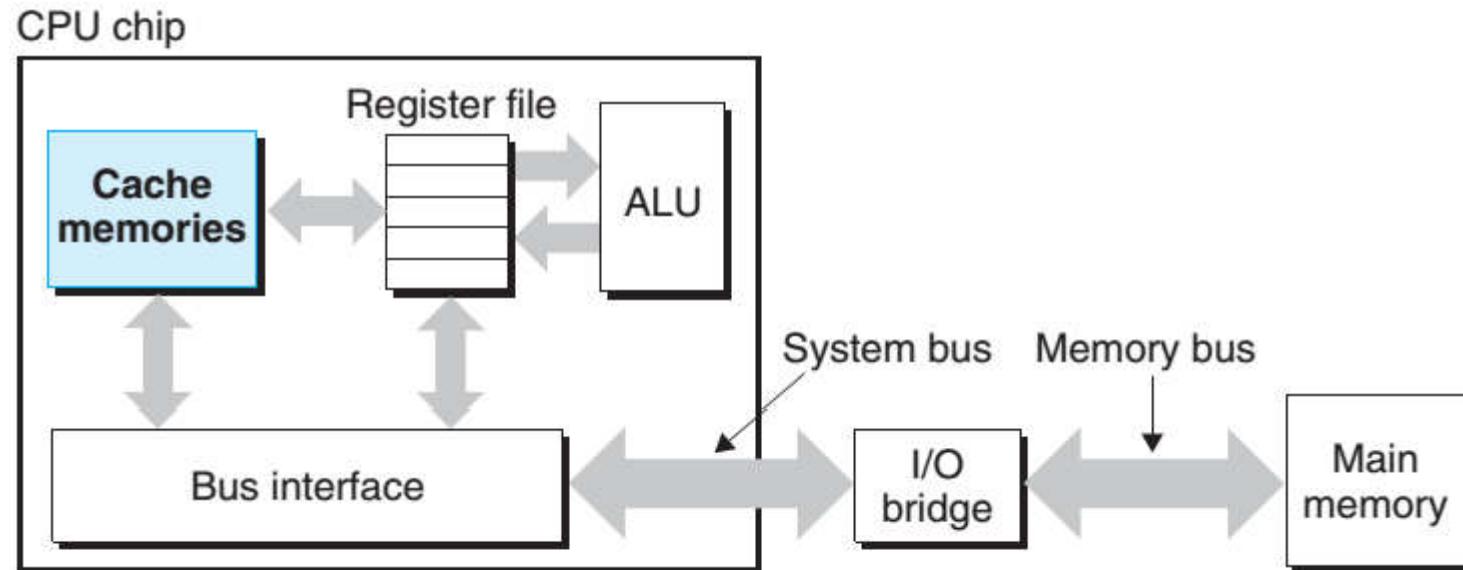
- ▶ Writing output “String”



Outline

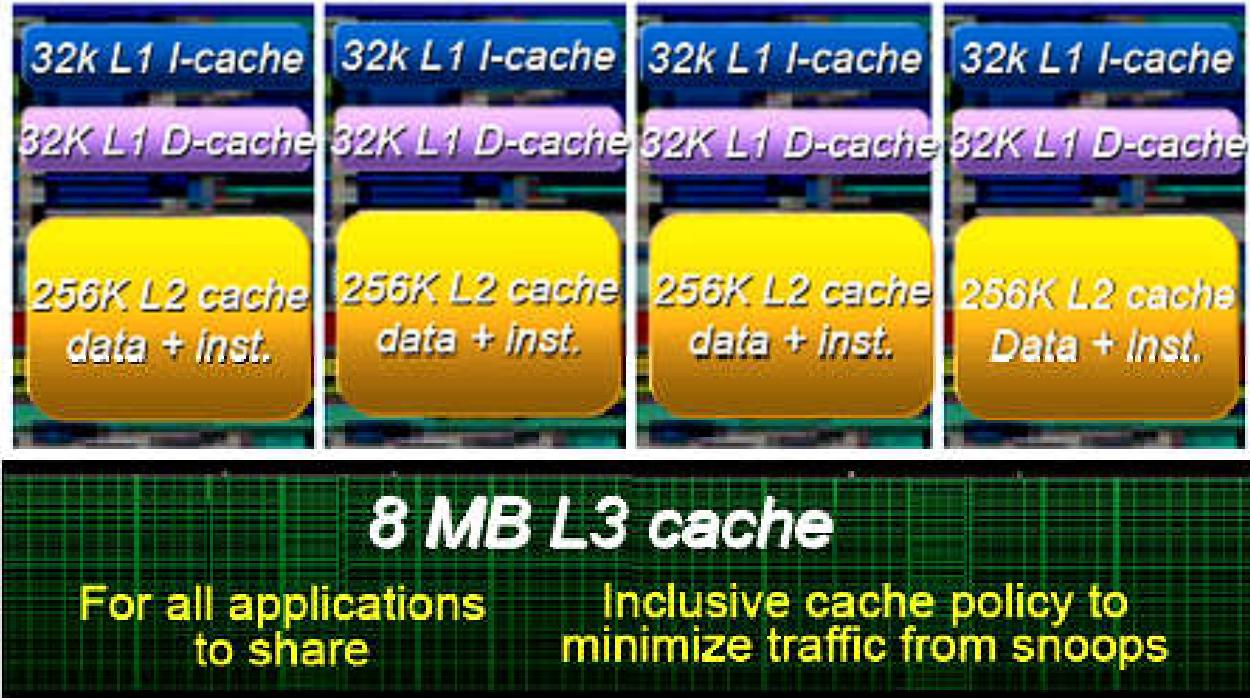
- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

Cache Memory



Cache Memory (contd.)

▶ Intel Core i7



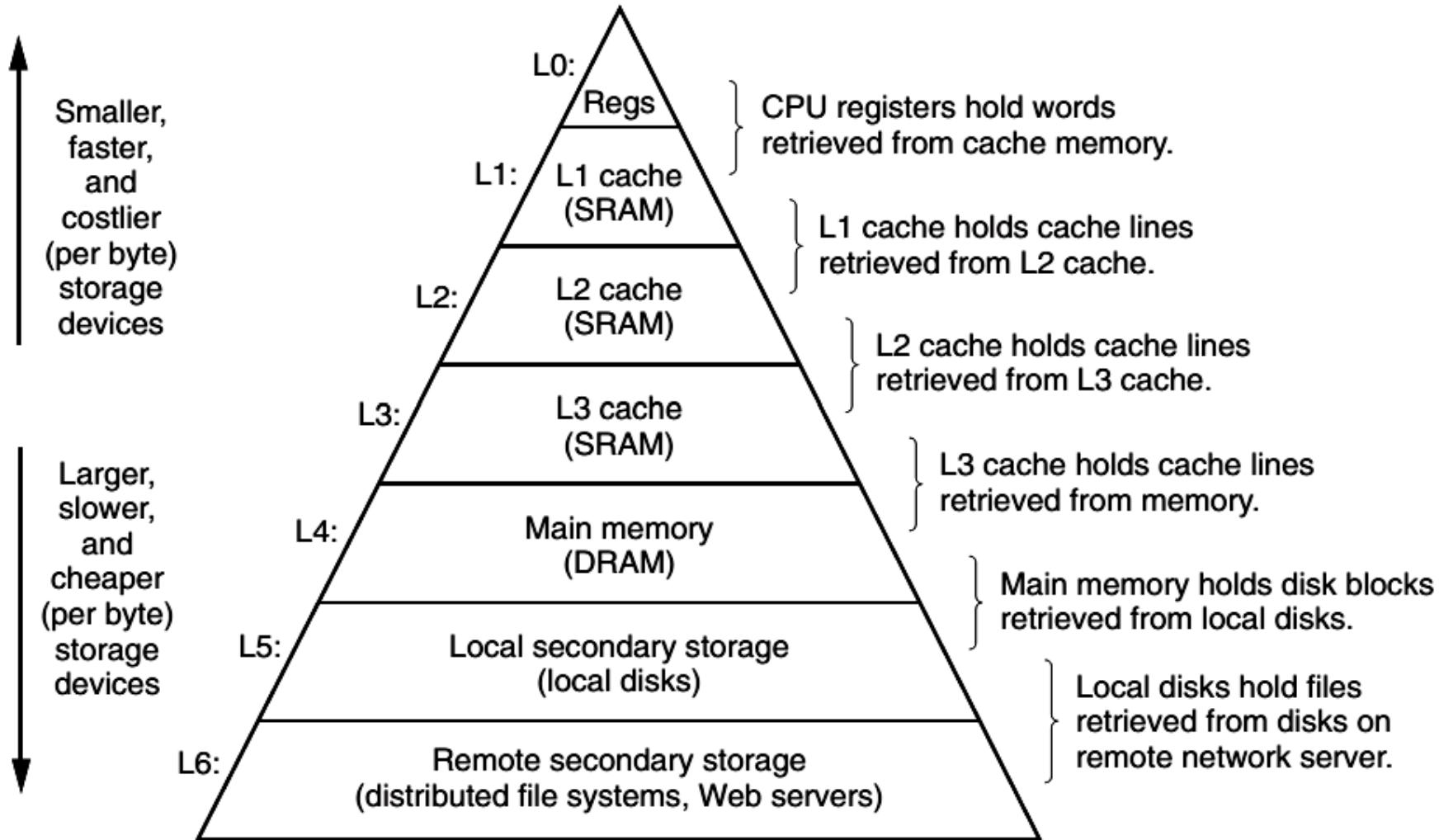
Cache Memory (contd.)

- ▶ Cache: L1
 - As fast as the Registers
- ▶ Cache: L2
 - 5–10 times faster than main memory
- ▶ Cache: L3
 - About 2-times faster
- ▶ All types are implemented using SRAM

Outline

- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

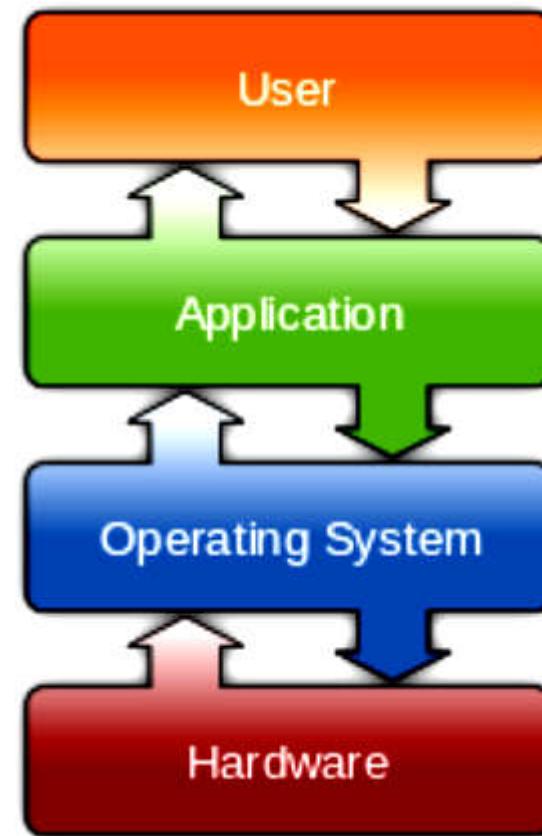
Storage Hierarchy



Outline

- ▶ Revisit C Compilation
- ▶ Tour of a Computer System
- ▶ Running a C program
- ▶ Cache Memory
- ▶ Storage Hierarchy
- ▶ Operating System Concepts

Operating System Concepts

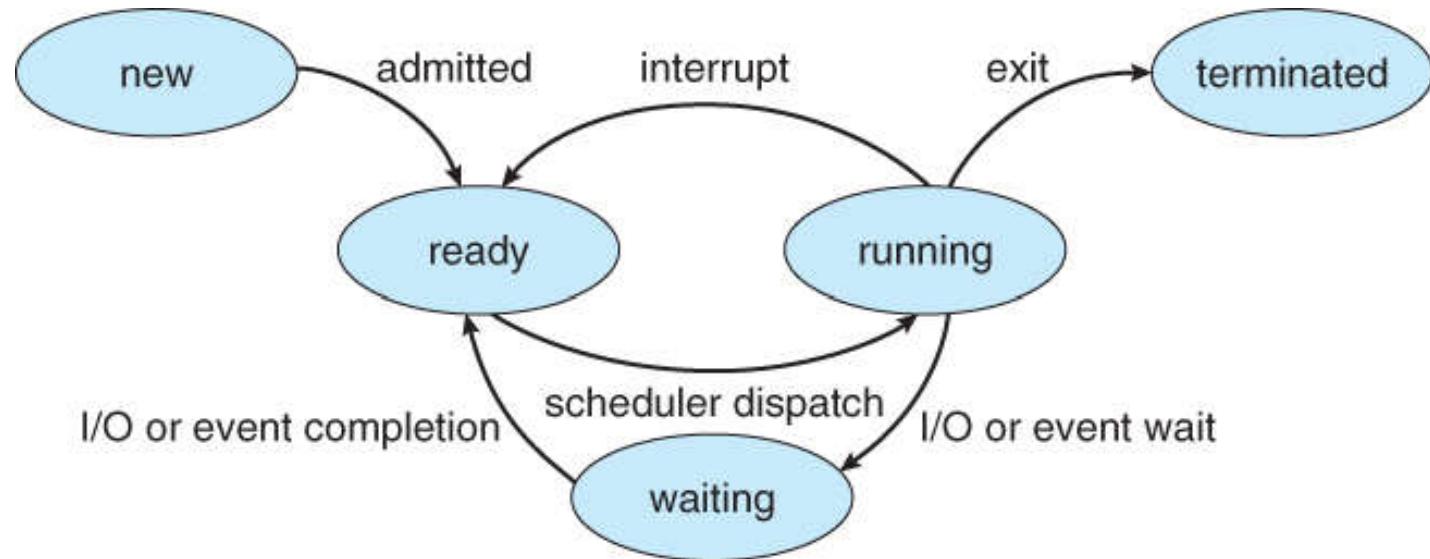


Operating System Concepts (contd.)

- ▶ Single-Tasking
- ▶ Multi-Tasking
- ▶ Multi-User / Time-Shared
- ▶ Real-Time
- ▶ Distributed
- ▶ Embedded

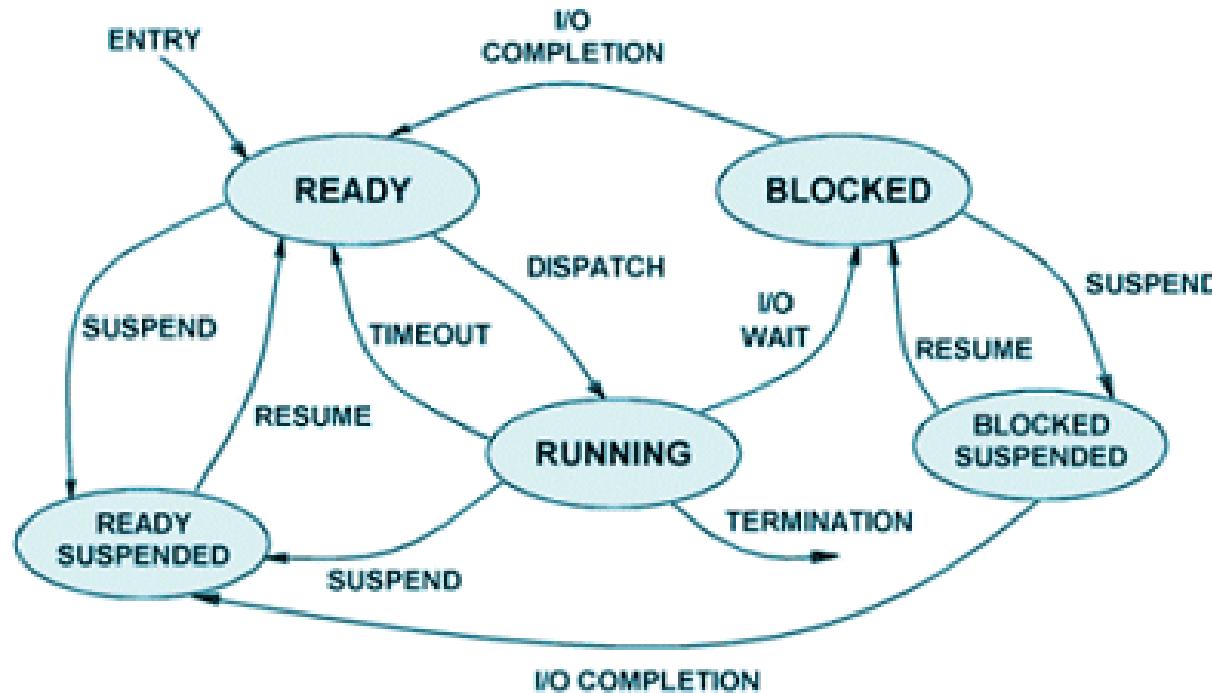
Operating System Concepts (contd.)

▶ Process States

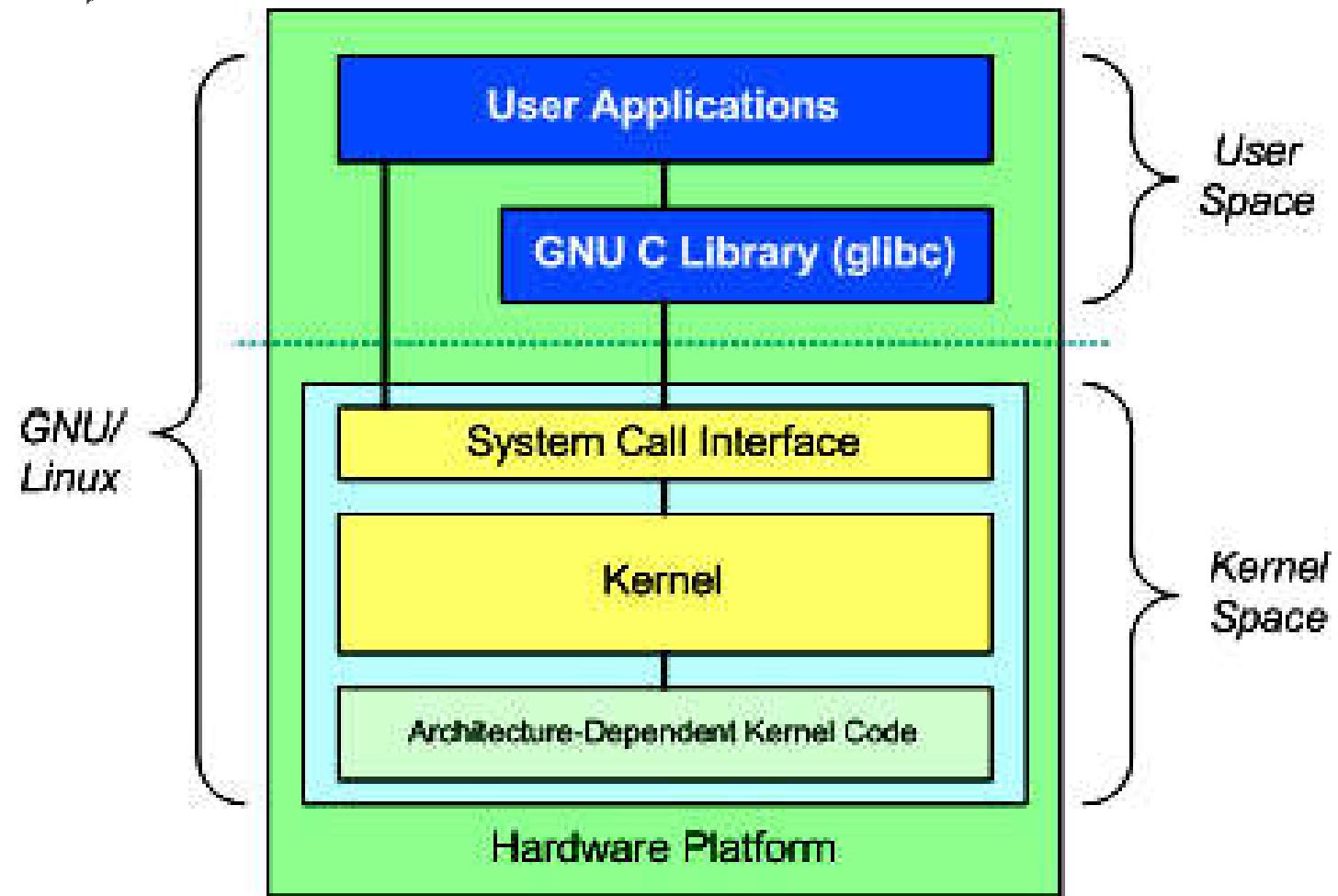


Operating System Concepts (contd.)

▶ Process States (advanced)

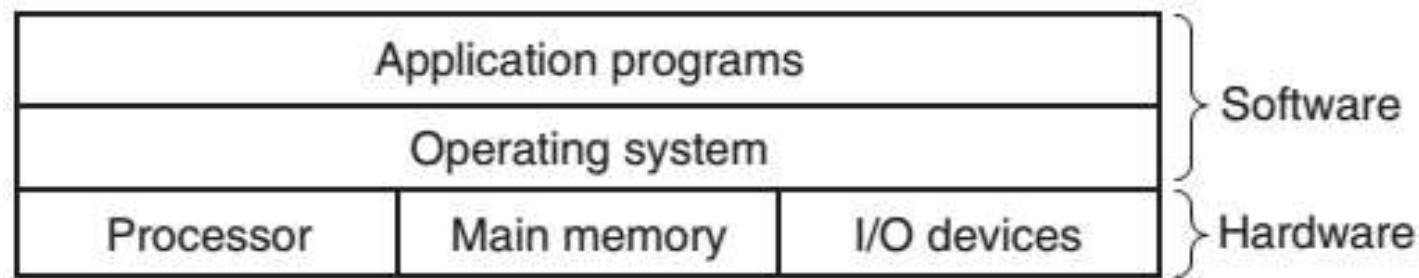


Operating System Concepts (contd.)

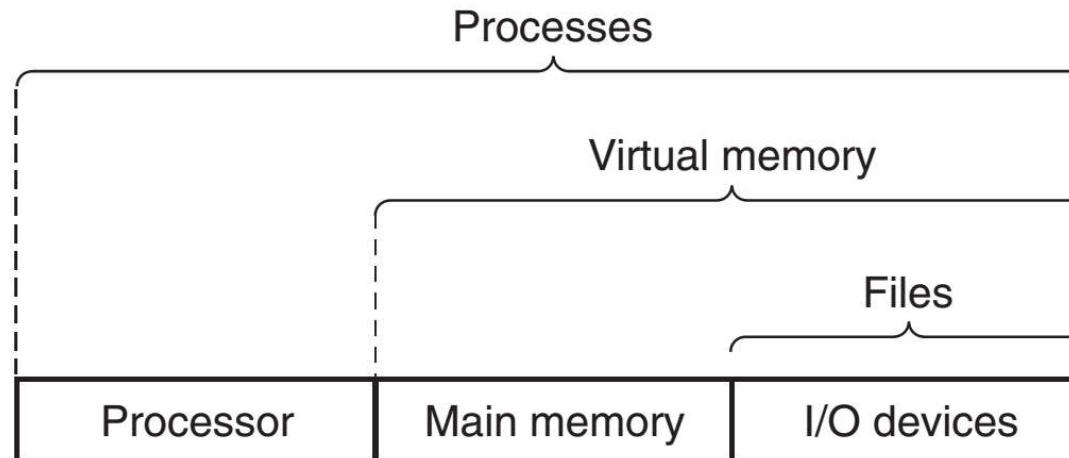


Operating System Concepts (contd.)

▶ Layered view



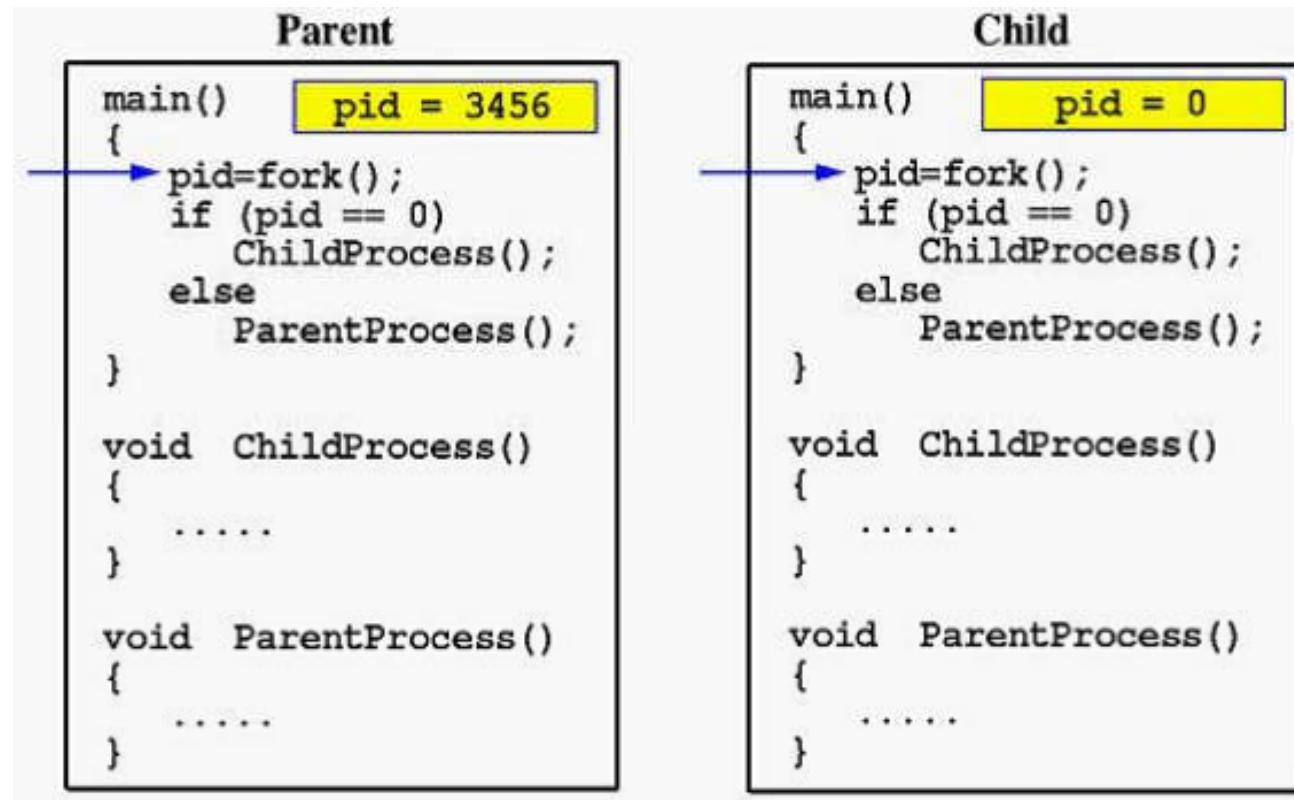
▶ Abstraction view



Operating System Concepts (contd.)

Process:

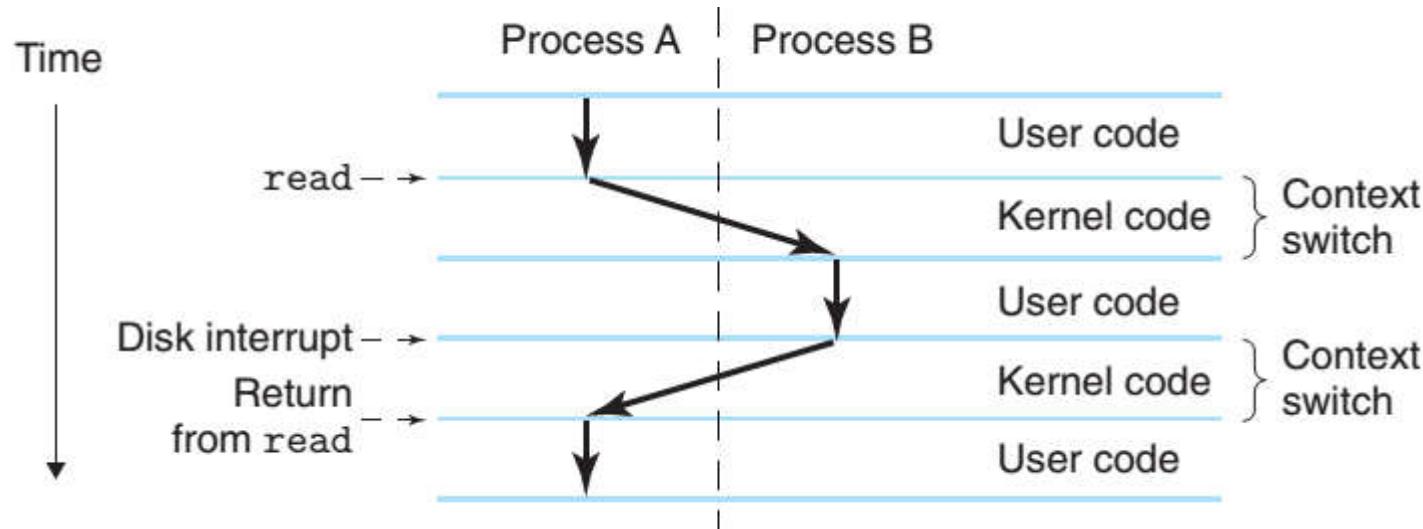
- ▶ Program in execution



Operating System Concepts (contd.)

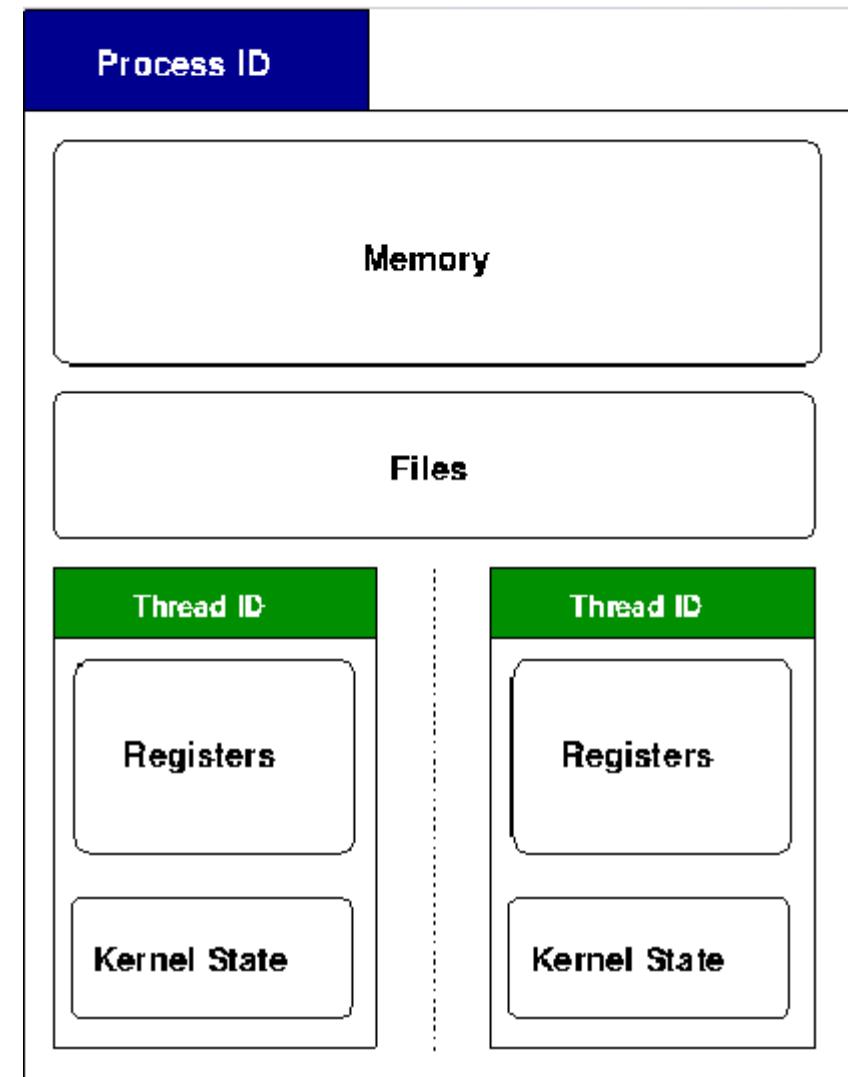
Process:

- ▶ Program in execution



Operating System Concepts (contd.)

▶ Threads



Operating System Concepts (contd.)

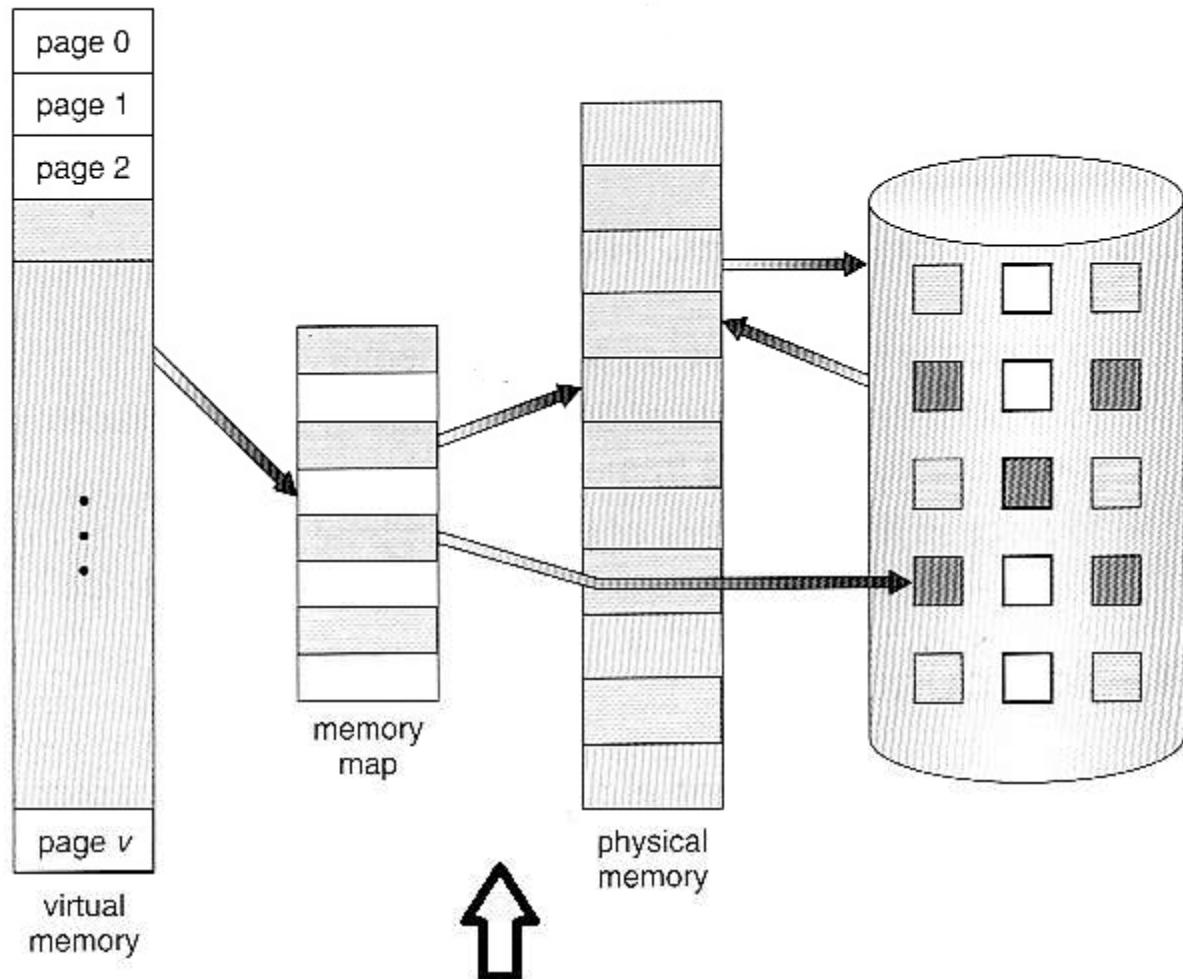
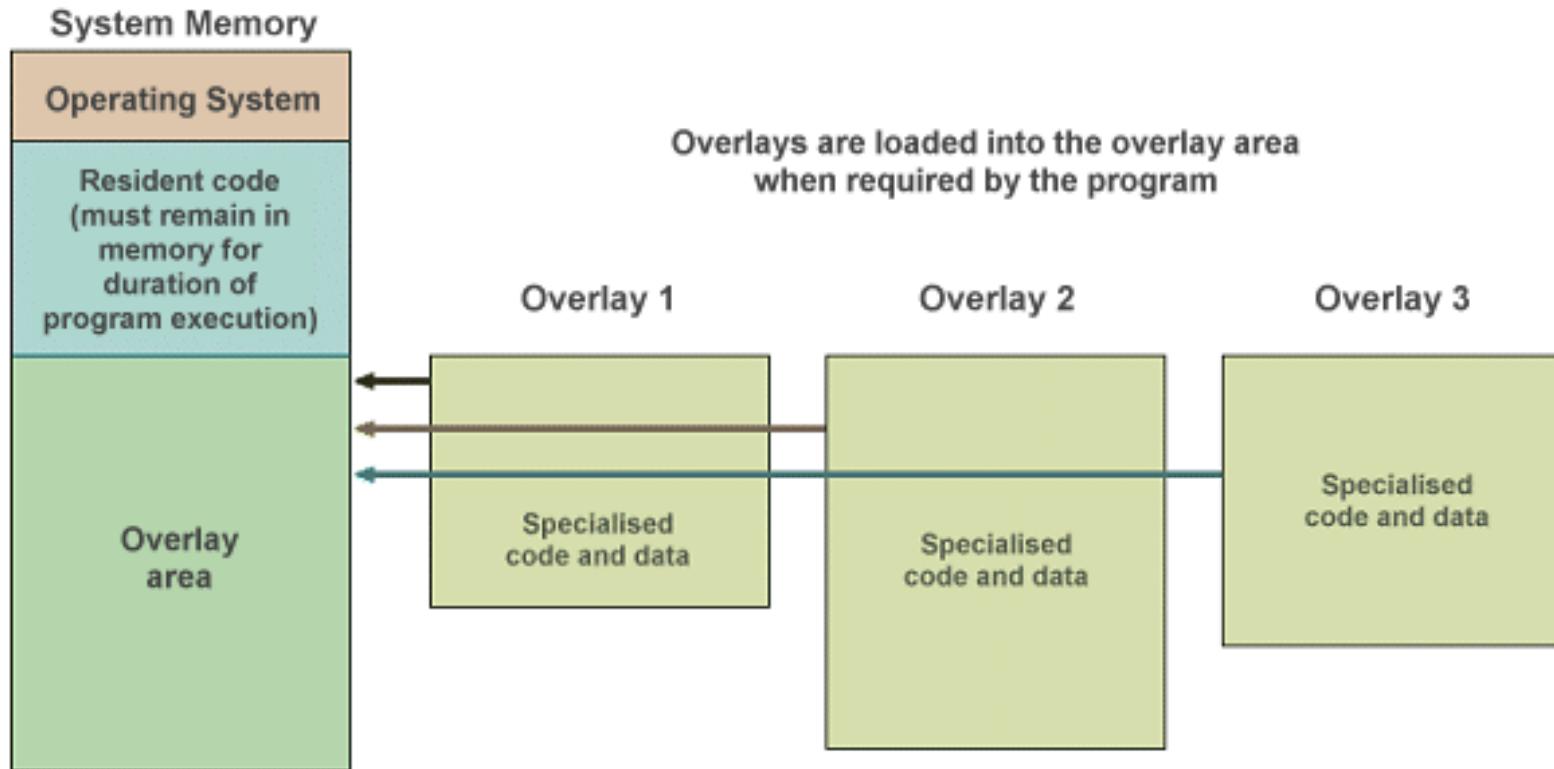


Diagram showing virtual memory that is larger than main memory

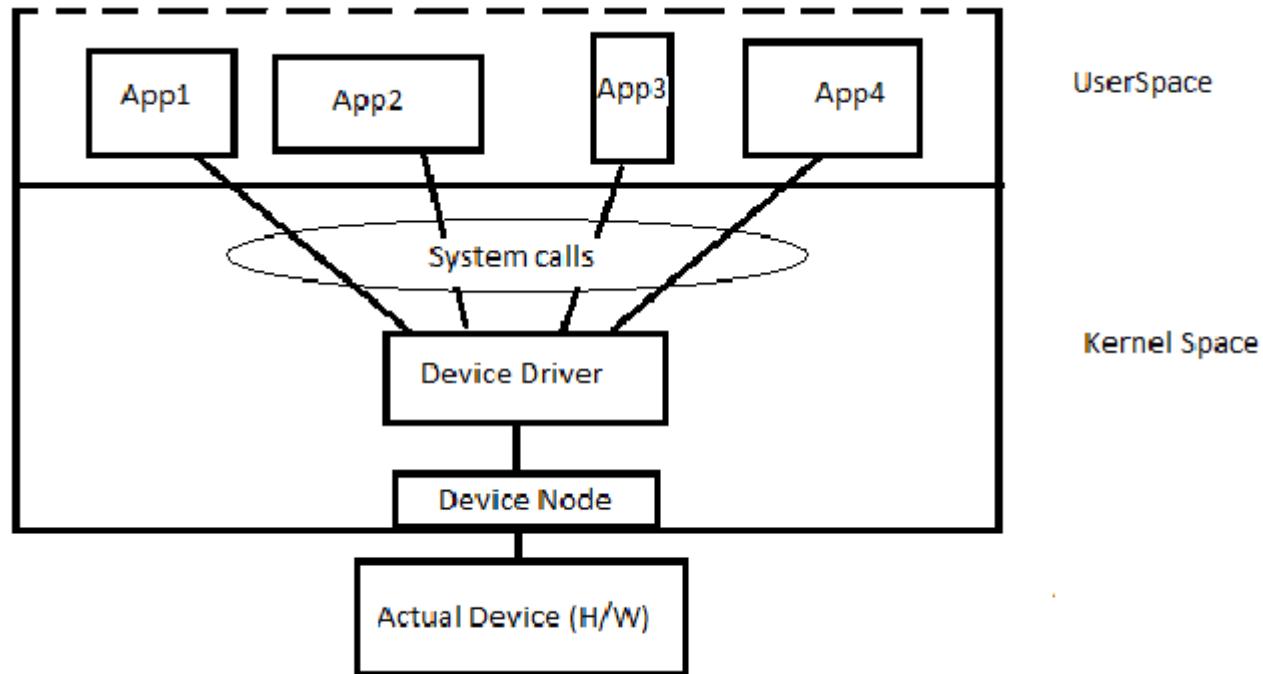
Operating System Concepts (contd.)



Operating System Concepts (contd.)

Files:

- ▶ Files are **FILES**
- ▶ Devices are also **FILES**



Operating System – Tasks

- ▶ Memory management
- ▶ Device management
- ▶ Processor management
- ▶ I/O programs
- ▶ File systems
- ▶ Searching / sorting
- ▶ Scheduler
- ▶ Libraries

Other System Software

- ▶ Compiler-compiler
- ▶ Cross compiler
- ▶ Cross assembler
- ▶ Emulator
- ▶ Preprocessor
- ▶ Macro-processor
 - MASM, TASM, VAX

Programming Considerations

- ▶ Development and Production environments
- ▶ Making Software Portable
- ▶ Software over Internet
- ▶ Programs as Components
- ▶ Quick-and-Dirty Programming
- ▶ Dynamic/Flexible/Adaptive Software

Take Away

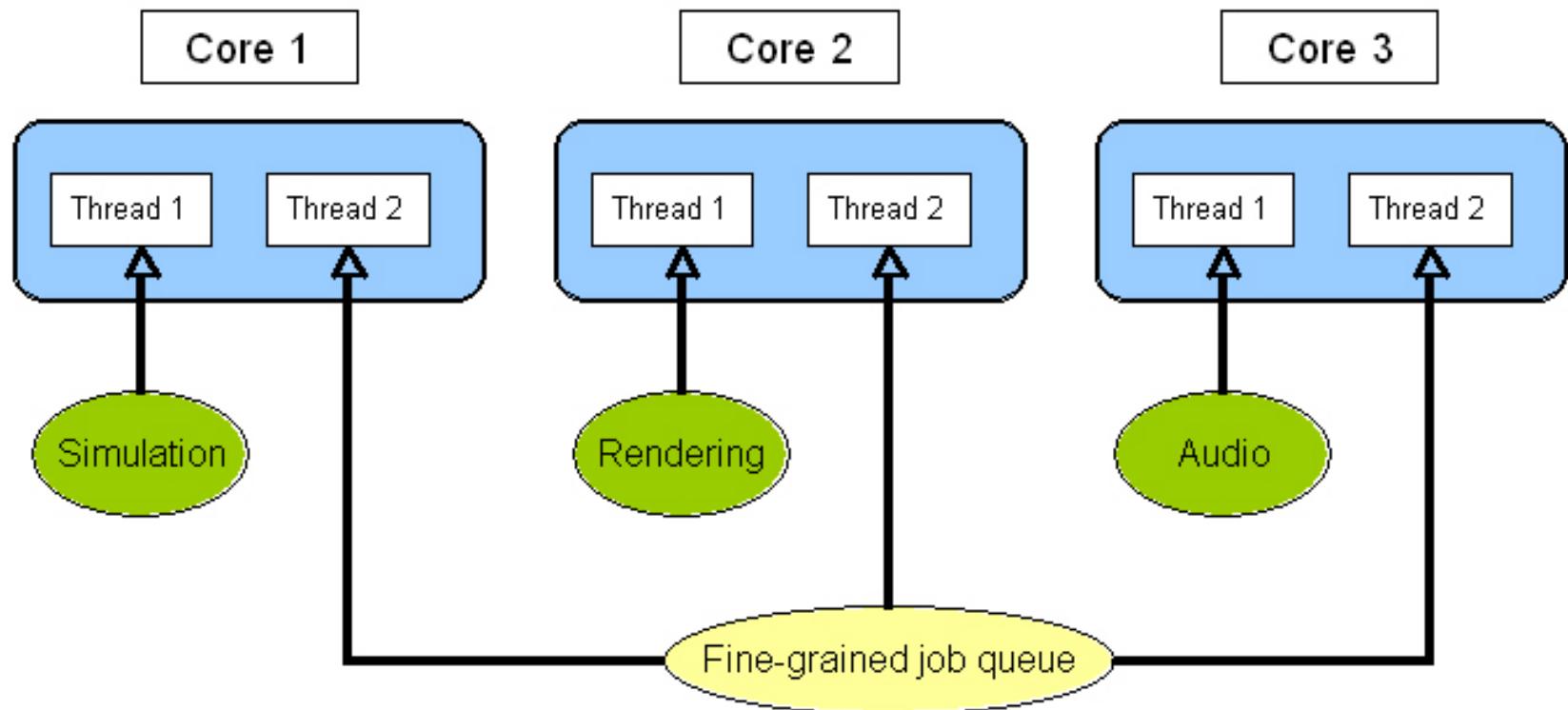
- ▶ Concurrency
 - Multiple simultaneous activities
- ▶ Parallelism
 - Concurrency to make systems run faster

Take Away (contd.)

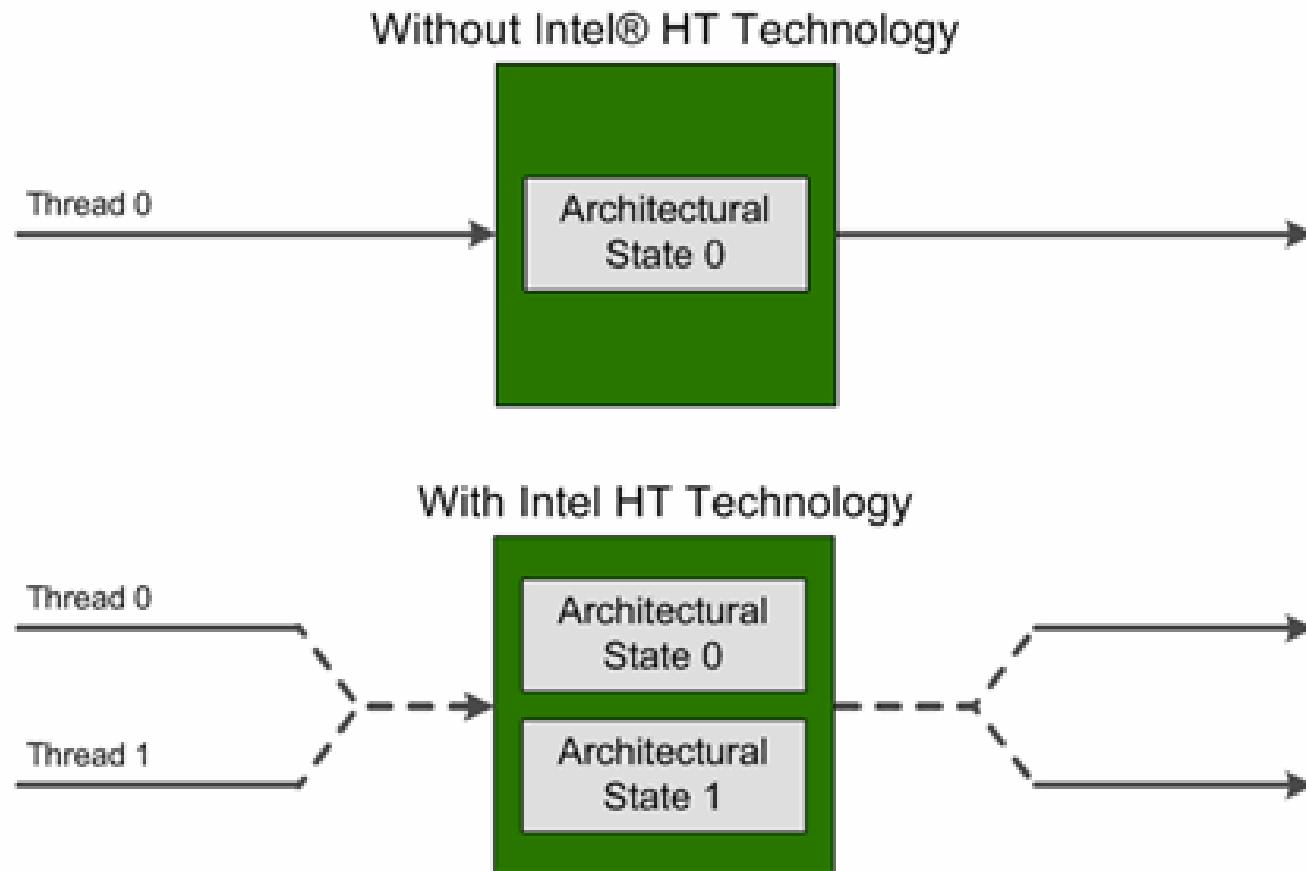
Hardware Threads:

- ▶ Thread-level Concurrency
 - Uni-processor
 - Multi-processor
- ▶ Hyper-threading
 - Simultaneous multithreading
 - Multiple: PC, other registers
 - Single: ALU, FPU

Take Away (contd.)



Take Away (contd.)



Take Away (contd.)

Instruction-level Concurrency:

- ▶ Previous Systems:
 - 1 instruction takes 3–4 Machine Cycles
- ▶ Superscalar
 - System that can execute more than ONE instructions per Cycle