Welcome to CS 241
Systems Programming at Illinois

Robin Kravets and
Brighten Godfrey

The Team

- Robin Kravets
  - Office: 3114 SC
  - Office Hours
    - 11 – 12 Thursdays
  - *rhk@cs.uiuc.edu*
- Brighten Godfrey
  - Office: 3128 SC
  - Office Hours
    - 2 – 3 Wednesdays
  - pbg@illinois.edu
- TAs
  - Wade Fagen, Liping Chen, Farhana Ashraf, Riccardo Crepaldi
  - Office hours and locations on website
- Discussion Sections
  - 8 sessions (Thursdays 9, 10, 11, 12, 1, 2, 3, 4)
  - All sections in SC 0220

News and Email

- news
  - class.cs241
    - All class questions
    - This is your one-stop help-line!
    - Will get answer < 24 hours
  - class.cs241.announce
    - All class announcements (staff only)
- e-mail
  - cs241help-sp10@cs.uiuc.edu
  - Personal questions not postable on the news group

The Textbook

- Introduction to Systems Concepts and Systems Programming
  - University of Illinois Custom Edition
  - Copyright © 2007
  - Pearson Custom Publishing
  - ISBN 0-536-48928-9

- Taken from:
  - Operating Systems: Internals and Design Principles, Fifth Edition, by William Stallings
  - UNIX™ Systems Programming: Communication, Concurrency, and Threads, by Kay A. Robbins and Steven Robbins
  - Computer Systems: A Programmer's Perspective, by Randal E. Bryant and David R. O'Hallaron

Your CS 241 "Mission"

- Come to class
  - MWF, 11-11:50am
  - Please participate actively…
  - Attend 1 discussion section per week
- Read textbook
  - Reading assignments posted on webpage
- Homework (3)
- Programming assignments (7)
  - Group size specified per MP
- Midterm (March 8th in class)
- Final (7:00–10:00 PM, Monday, May 10)

Grading

Final Exam:	30%
Mid-term Exam:	20%
Homework (three total):	15%
Machine Problems (7 total):	30%
Participation:	5%
○ Lecture quizzes	

Deadlines

- Homework
  - Deadlines are strict
  - Late submissions will not be considered
- MPs
  - Please respect posted deadlines to ensure quick grading
  - Late MPs will be penalized 2% for each late hour (rounded off to the higher hour)
  - No submissions past 48 hours

Regrades

- Within one week of posting of grades for a quiz, homework, MP or exam
- Regrades must be submitted in writing on a separate piece of paper
  - Please do not write on your quiz, homework, MP or Exam

Academic Honesty

- Your work in this class **must** be your own.
- If students are found to have collaborated excessively or to have blatantly cheated (e.g., by copying or sharing answers during an examination or sharing code for the project), **all** involved will at a minimum receive grades of 0 for the first infraction and reported to the academic office.
- Further infractions will result in failure in the course and/or recommendation for dismissal from the university.

What is cheating in a programming class?

- At a minimum
  - Copying code
  - Copying pseudo-code
  - Copying flow charts
- Consider
  - Did some one else tell you how to do it?
- Does this mean I can’t help my friend?
  - No, but don’t solve their problems for them

Course Questions

- What is an operating system?
- What is it for?
- How do I use it?
- What is concurrency?
- What is system programming?

Course Objectives

- By the end of this course, you should know about operating systems
  - Identify the basic components of an operating system
  - Describe their purpose
  - Explain how they function
- Use the system effectively
  - Write, compile, debug, and execute C programs
  - Correctly use system interfaces provided by UNIX (or a UNIX-like operating system)

General Course Outline

- Understand the Basics (week 1-2)
  - Use UNIX system calls correctly from within C programs
- Make the OS do tasks (week 3-8)
  - Create and manage processes and threads on UNIX
  - Control OS scheduling policy parameters
  - Exploit OS semaphores and mutexes
  - Enable inter-process communication
  - Manage shared memory
- Write networked applications (week 9-11)
  - Use communication protocols (TCP/IP) and interfaces (Sockets)
  - Write distributed multi-threaded apps that talk across a network
- Advanced machine resources (week 12-14)
  - Take advantage of OS signals and signal handlers
  - Set OS timers and clocks
  - File management
  - Memory allocation
- Advanced Topics (week 15-16)

General Course Outline

- Understand the Basics (week 1-2)
  - MP1 C Pointers and Strings
- Make the OS do tasks (week 3-8)
  - MP2 Processes and Threads
  - MP3 Scheduling
  - MP4 Synchronization

Midterm

  - Enable inter process
  - MP5 Inter-process Communication
  -

- Write networked applications (week 9-11)
  - MP6 Networking
  - Write distributed multi-threaded apps that talk across a network
- Advanced machine resources (week 12-14)
  - Take advantage of OS signals and signal handlers
  - Set OS timers and clocks
  - MP7 Files and I/O
  - Memory allocation
- Ad

Final

Complete Schedule

- See class webpage
- http://www.cs.uiuc.edu/class/cs241
  - Schedule is dynamic
  - Check regularly for updates

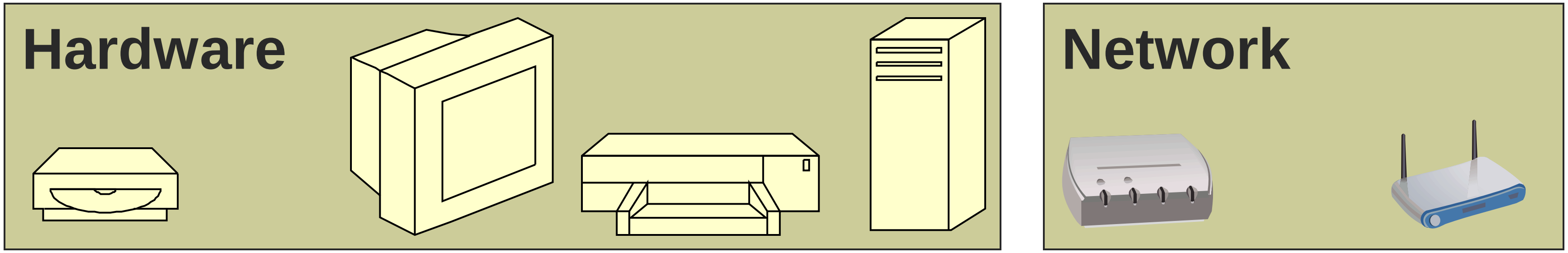
Your to-do List

- Visit the class webpage
  - Check out all the info
    - Especially schedule, grading policy, homework & MP hand-in instructions, and resources
  - http://www.cs.uiuc.edu/class/cs241
- Take a survey by Friday
  - On Compass: Assessments
- Familiarize yourself with newsgroups
  - see http://news.cs.uiuc.edu
  - Subscribe to: class.cs241 and class.cs241.announce
- Find a reference to refresh your C programming skills
  - http://www.lysator.liu.se/c/bwk-tutor.html

What is an operating system and why do I need one?

- What do we have?
  - Set of common resources
- What do we need?
  - A clean way to allow applications to use these resources!

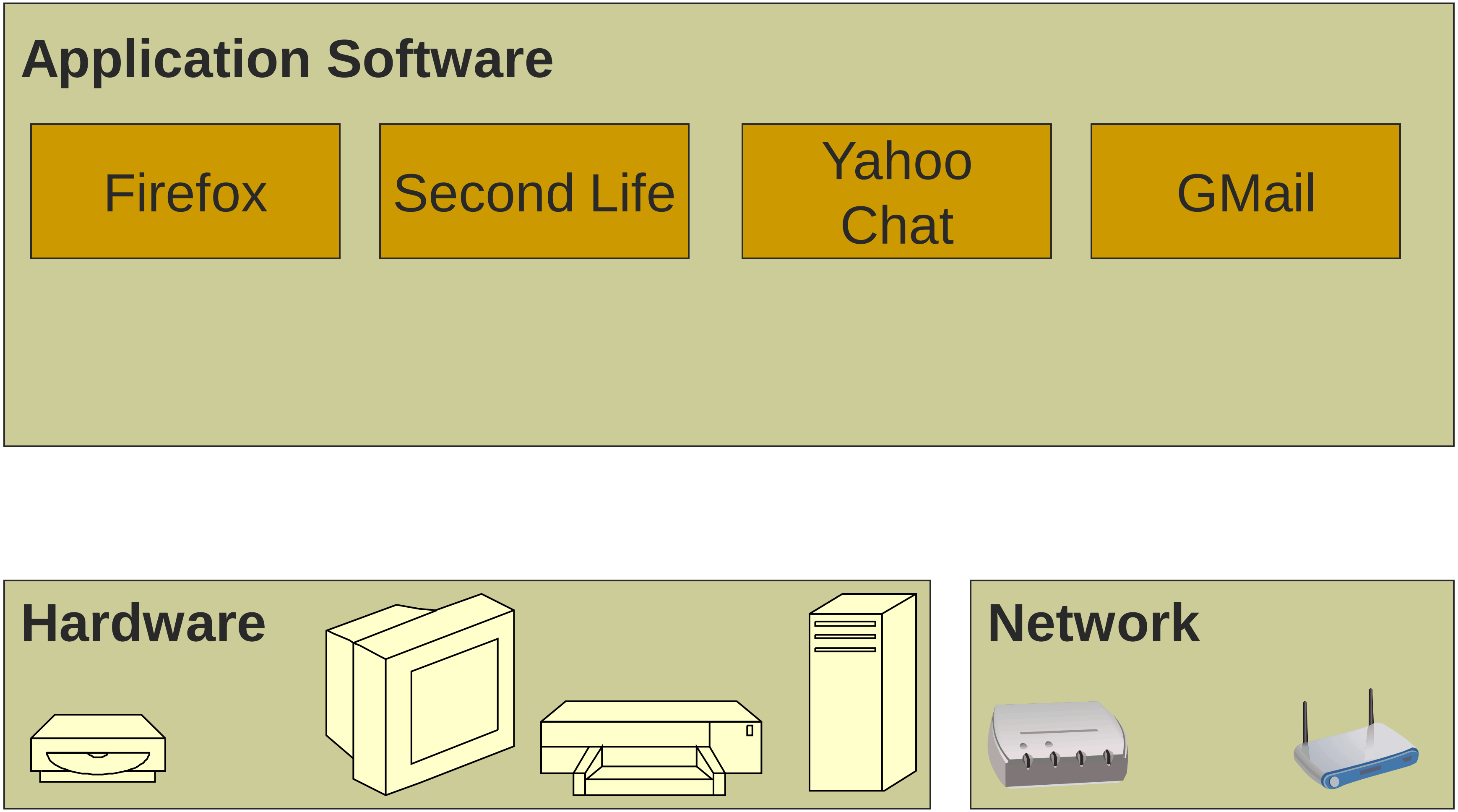
Resources



Hardware

Network

Applications



Application Software

Firefox

Second Life

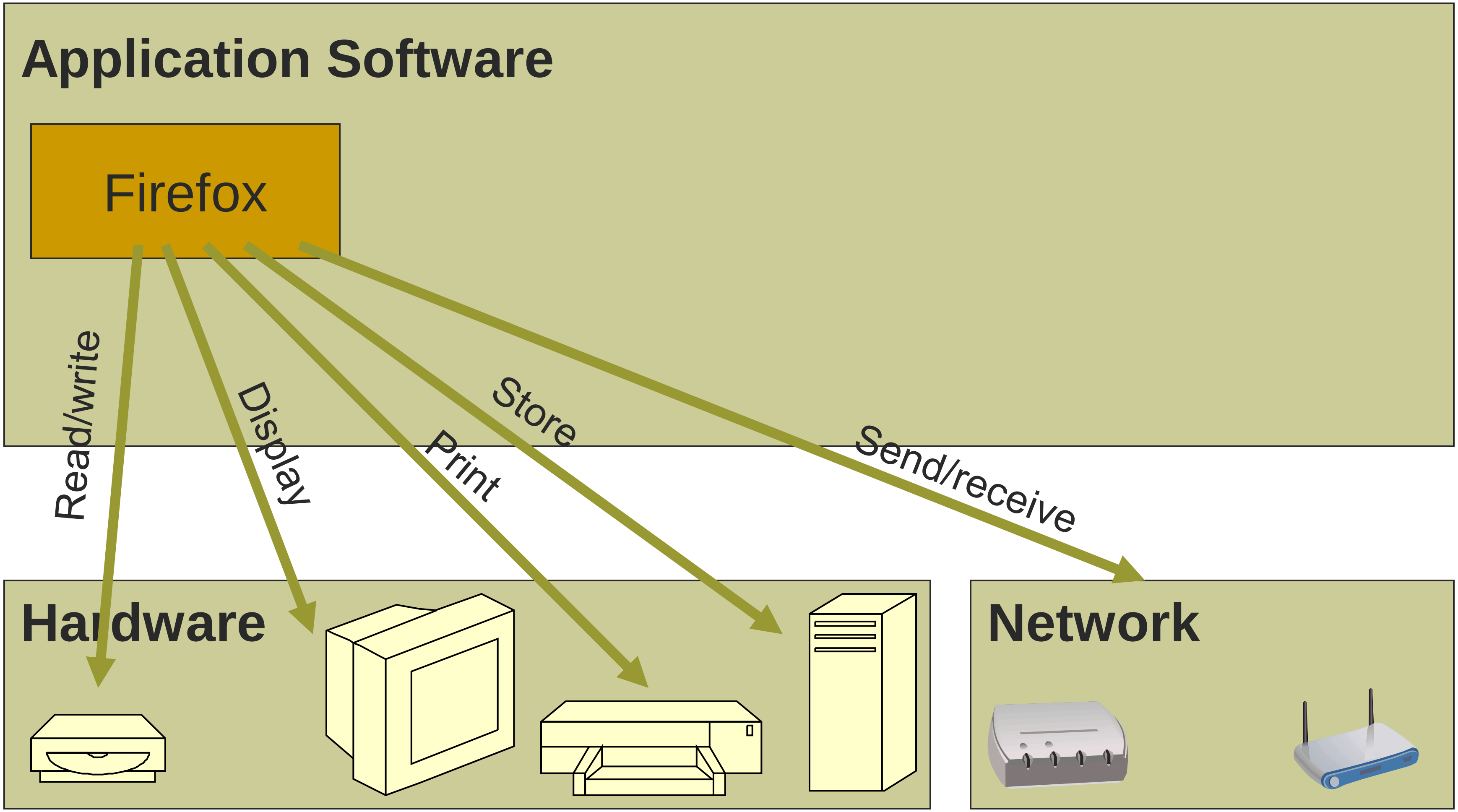
Yahoo Chat

GMail

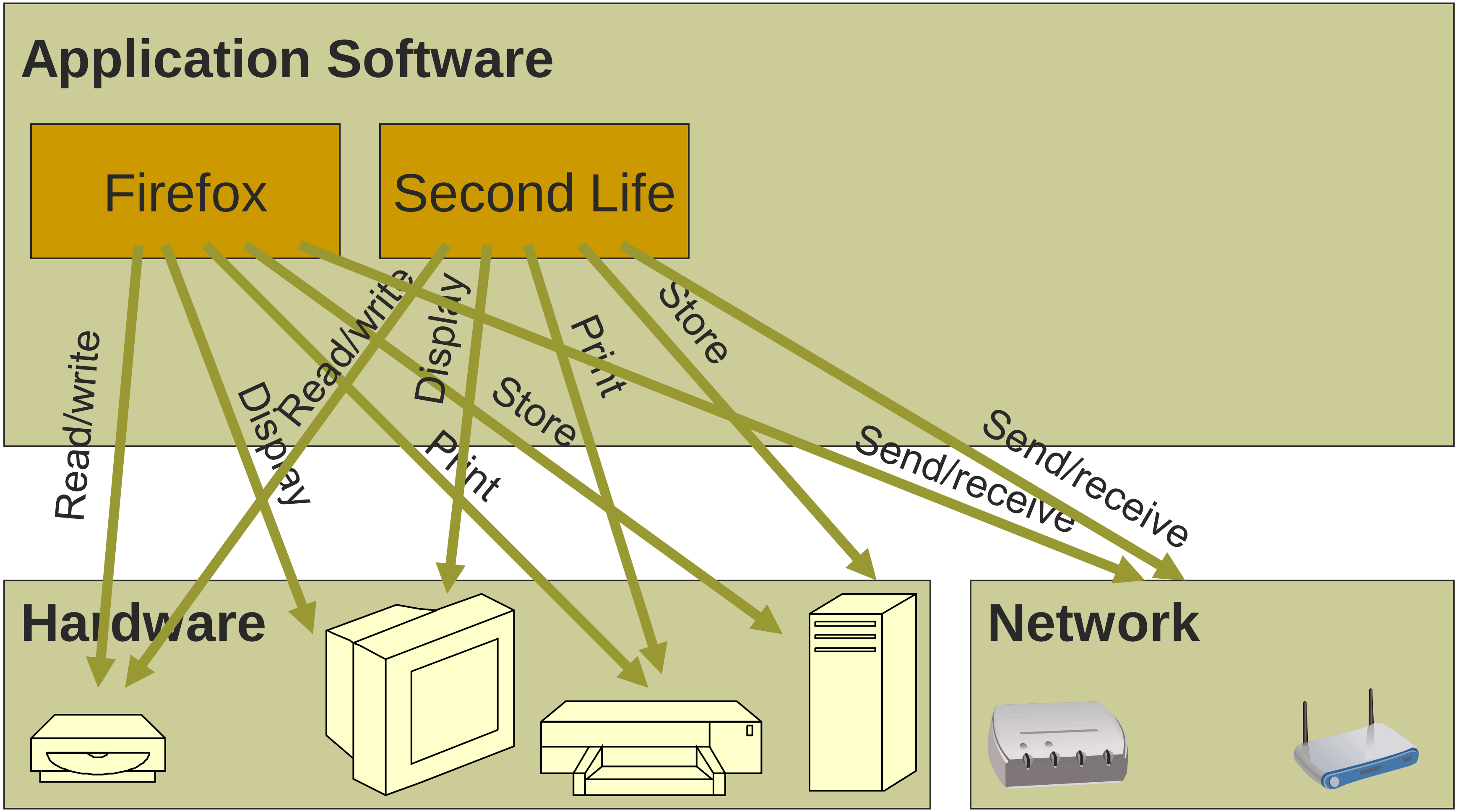
Hardware

Network

Application Requirements

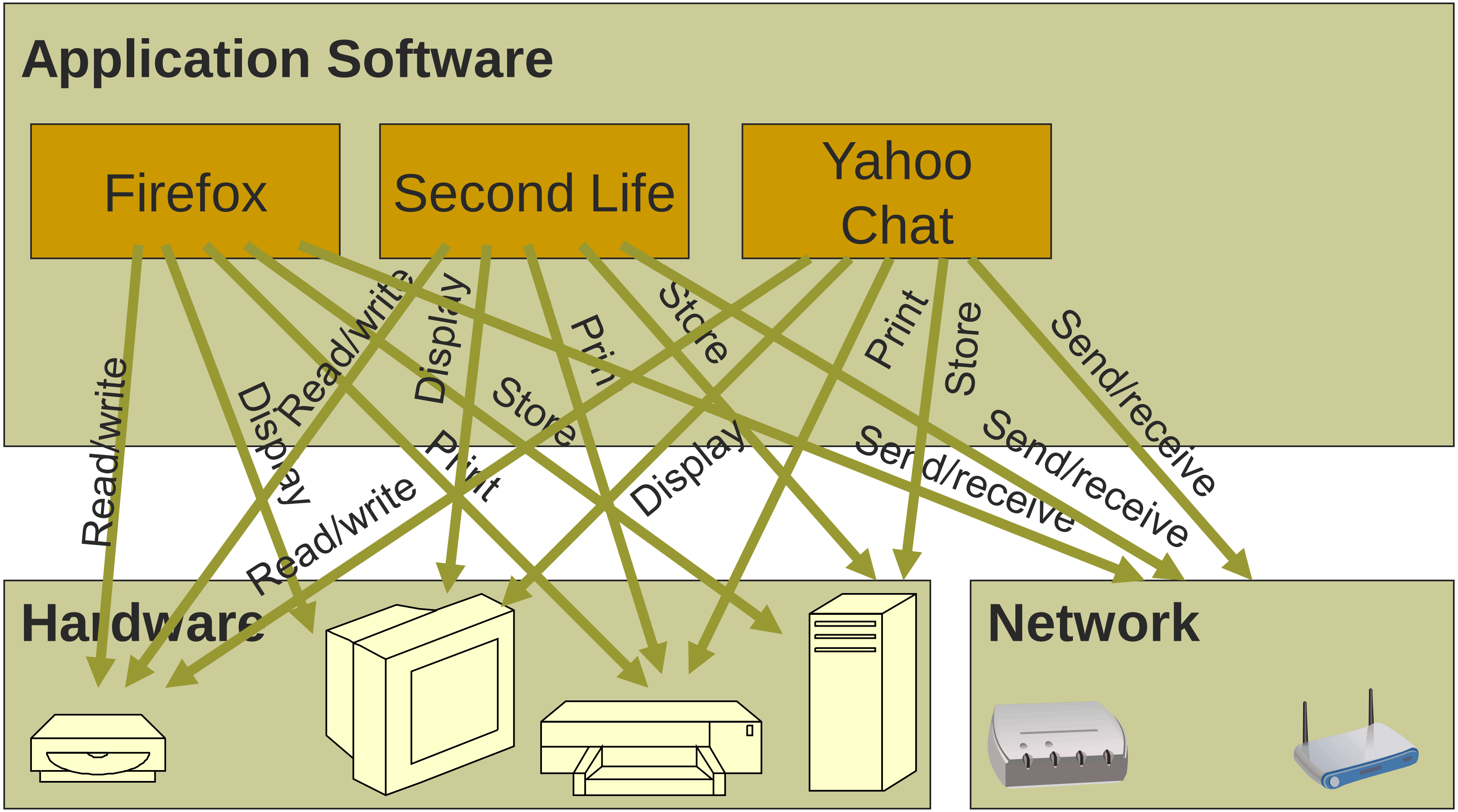


Two Applications?

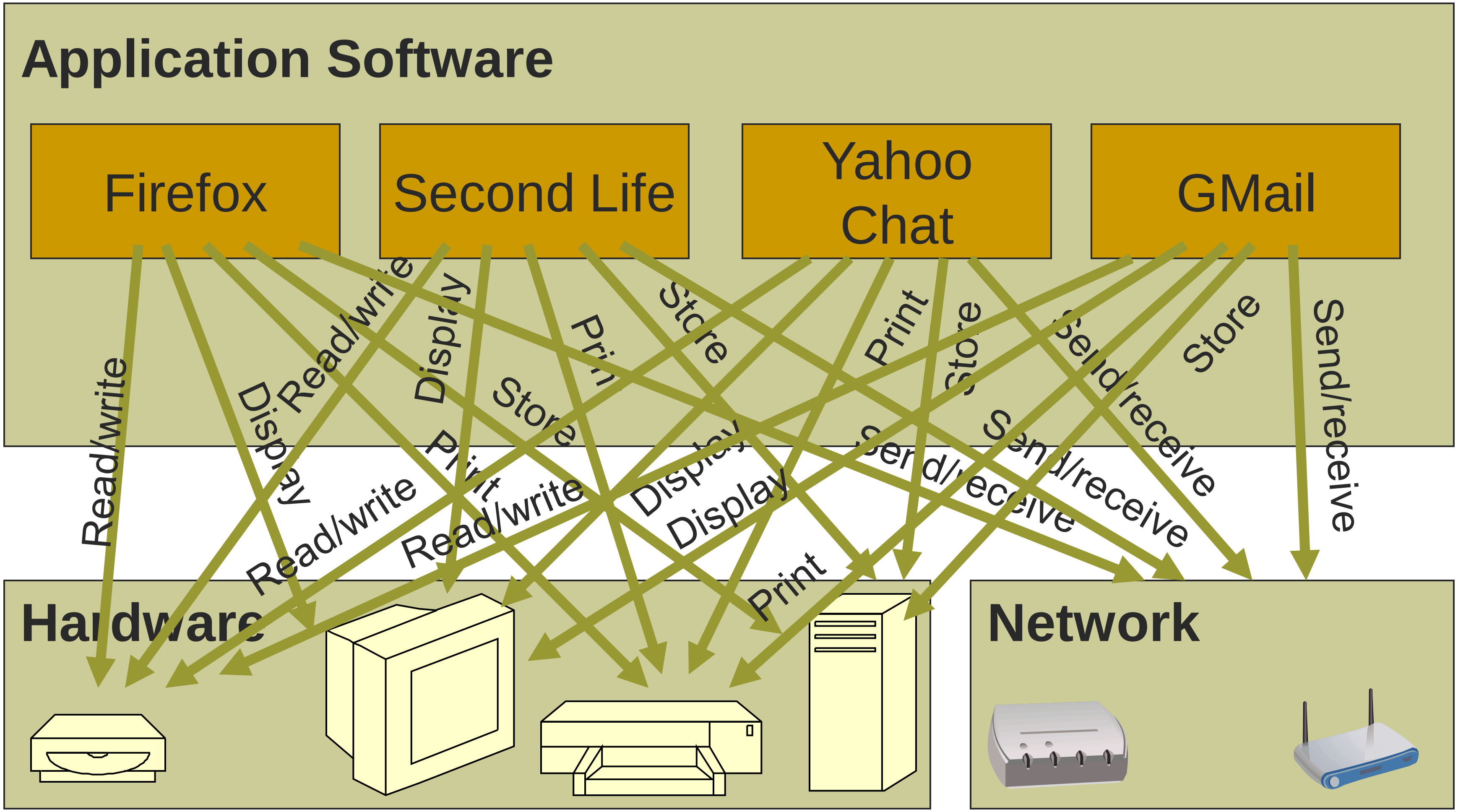


Copyright ©: University of Illinois CS 241 Staff

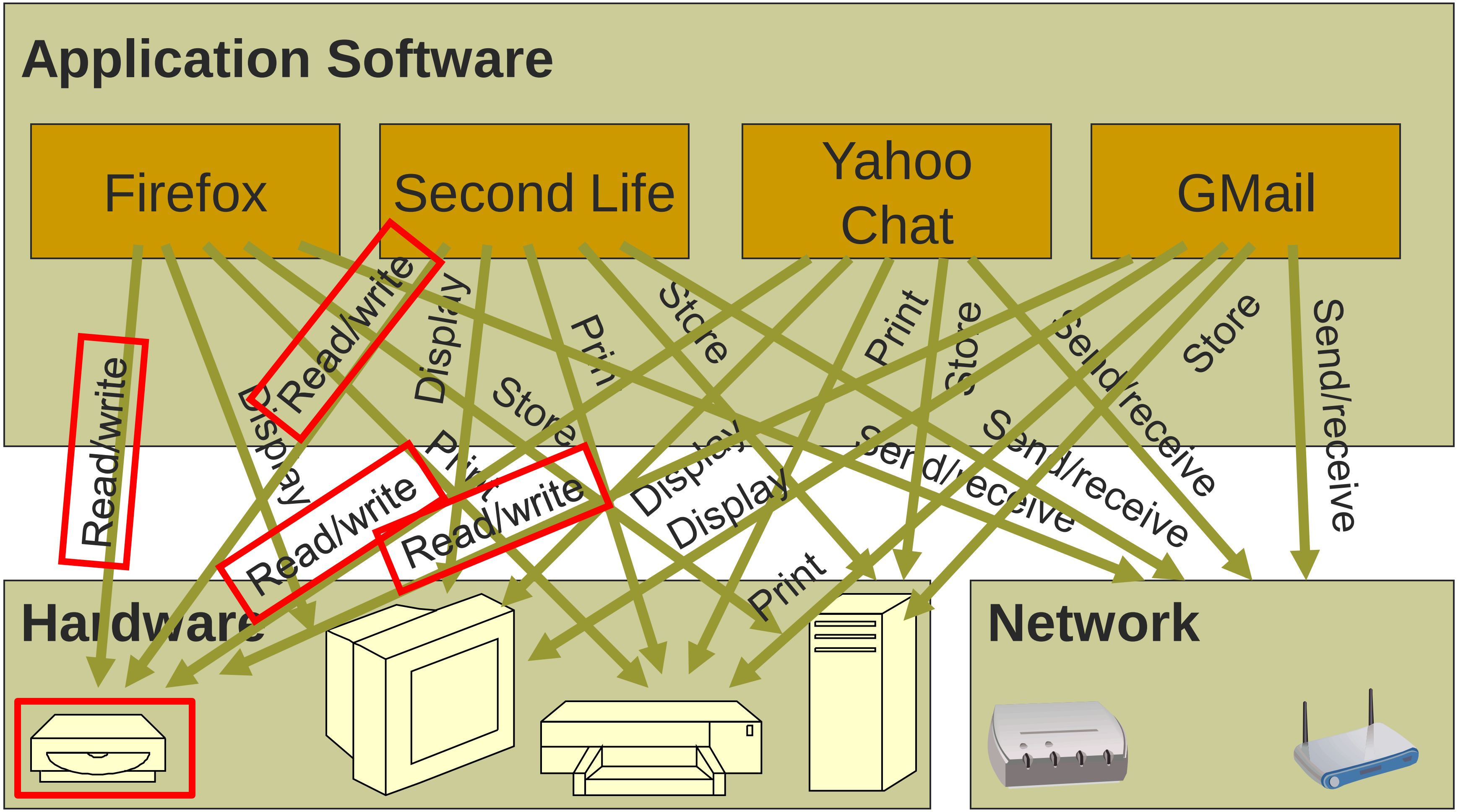
Managing More Applications?



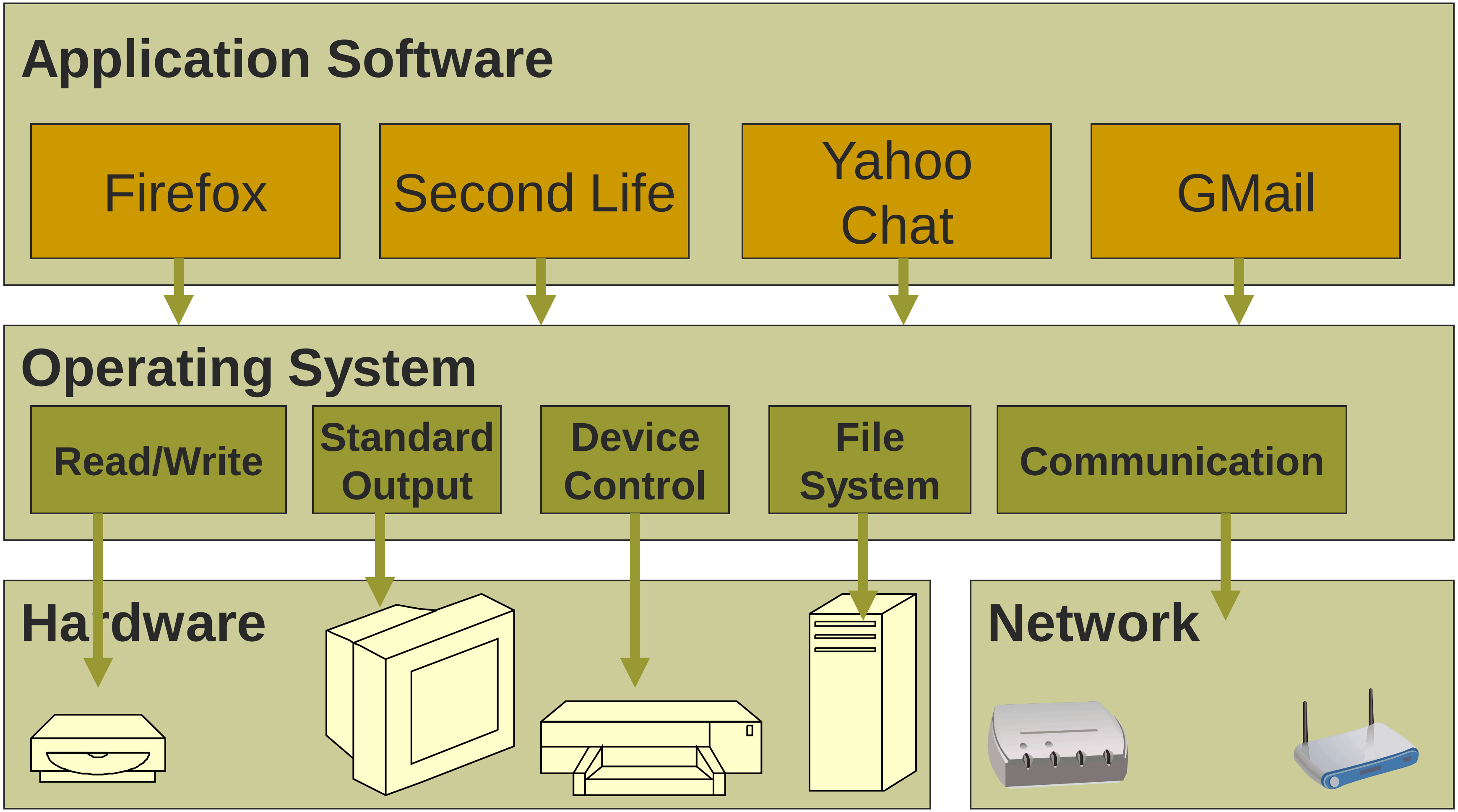
We need help!


Application Software
Firefox
Second Life
Yahoo Chat
GMail
Read/write
Display
Read/write
Display
Store
Print
Store
Print
Store
Send/receive
Store
Send/receive
Read/write
Read/write
Print
Display
Display
Send/receive
Send/receive
Print
Hardware
Network

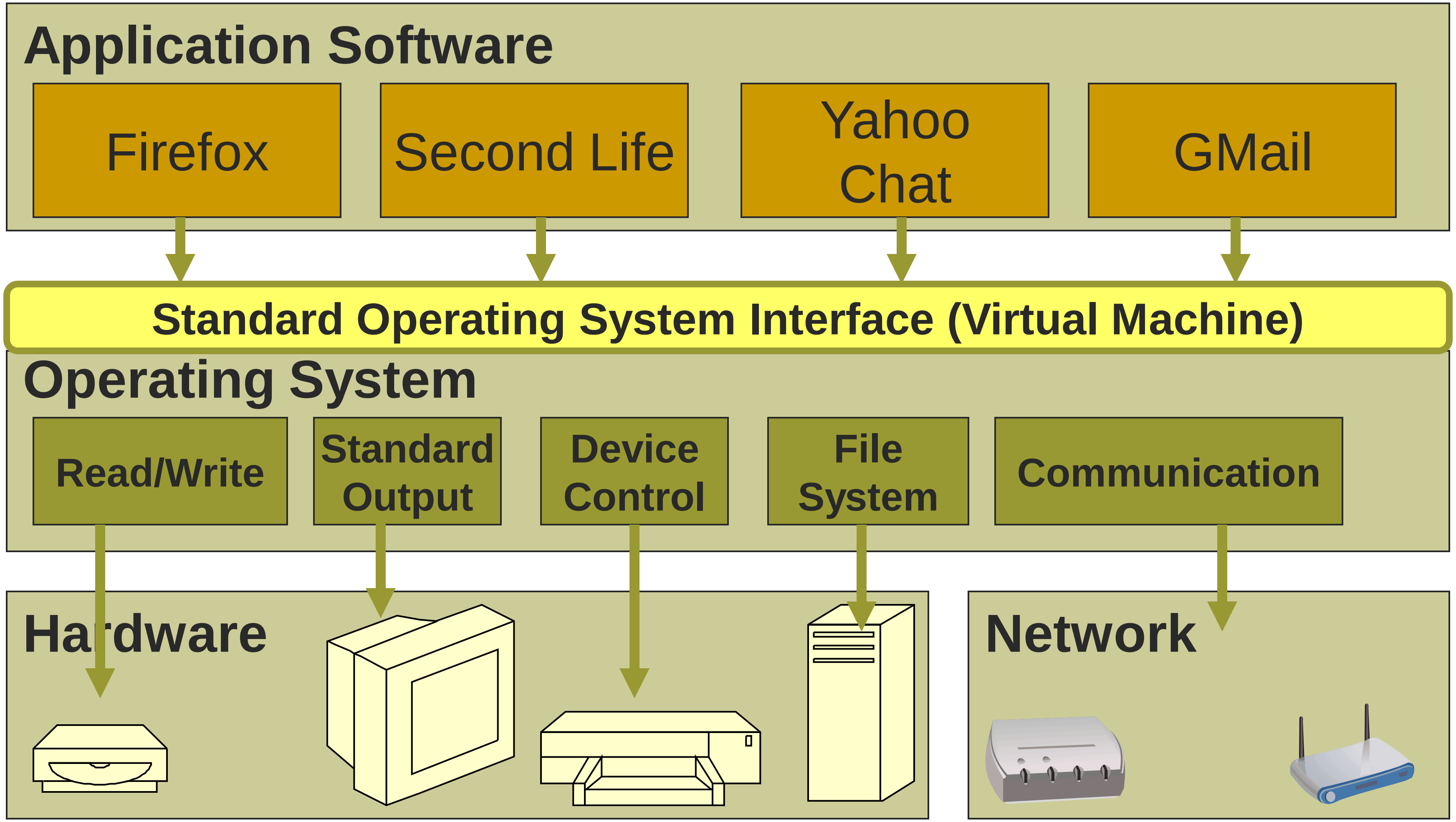
Approach: Find Common Functions



Delegate Common Functions



Export a Standard Interface



Application Software

Firefox

Second Life

Yahoo Chat

GMail

Standard Operating System Interface (Virtual Machine)

Operating System

Read/Write

Standard Output

Device Control

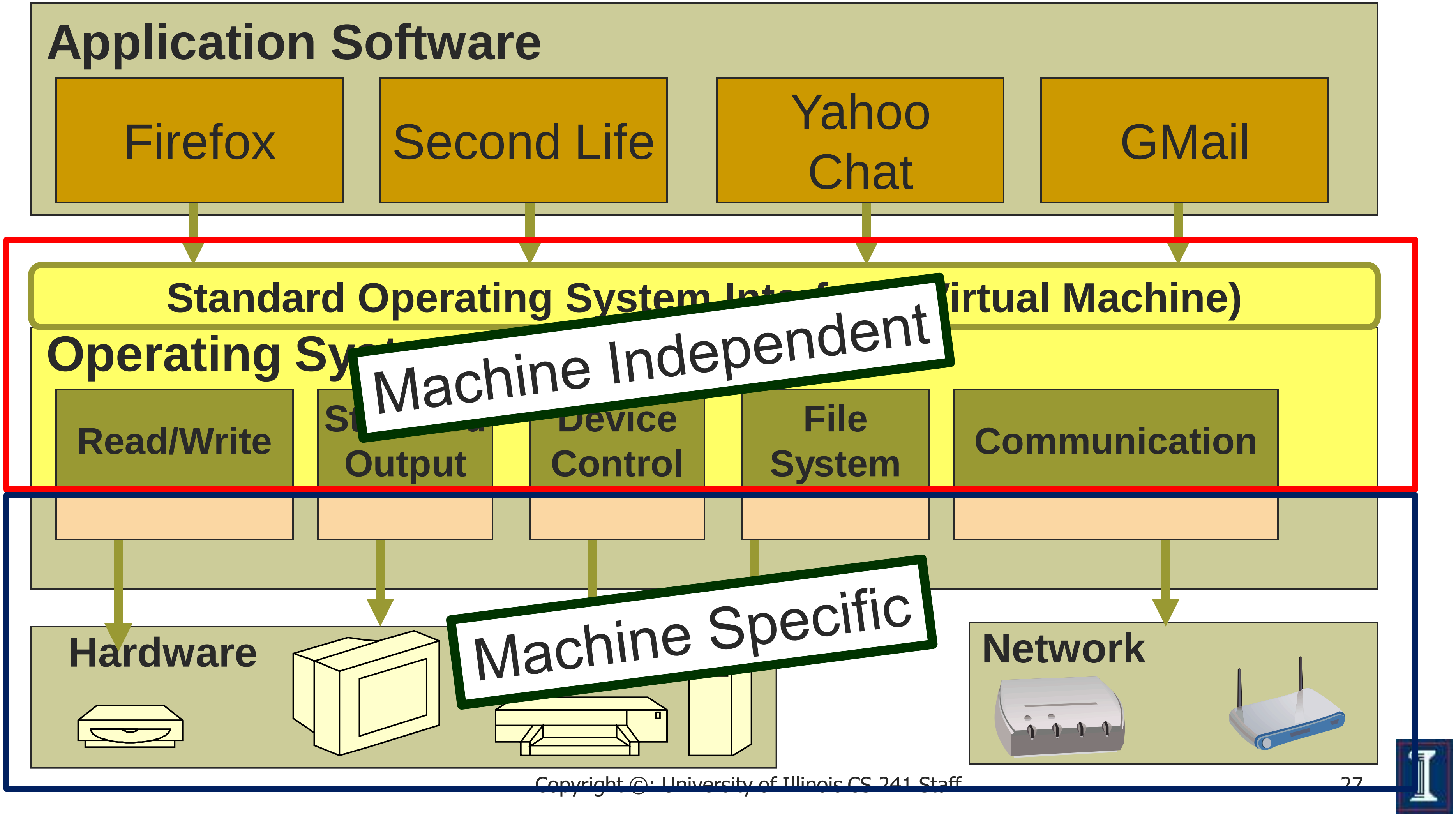
File System

Communication

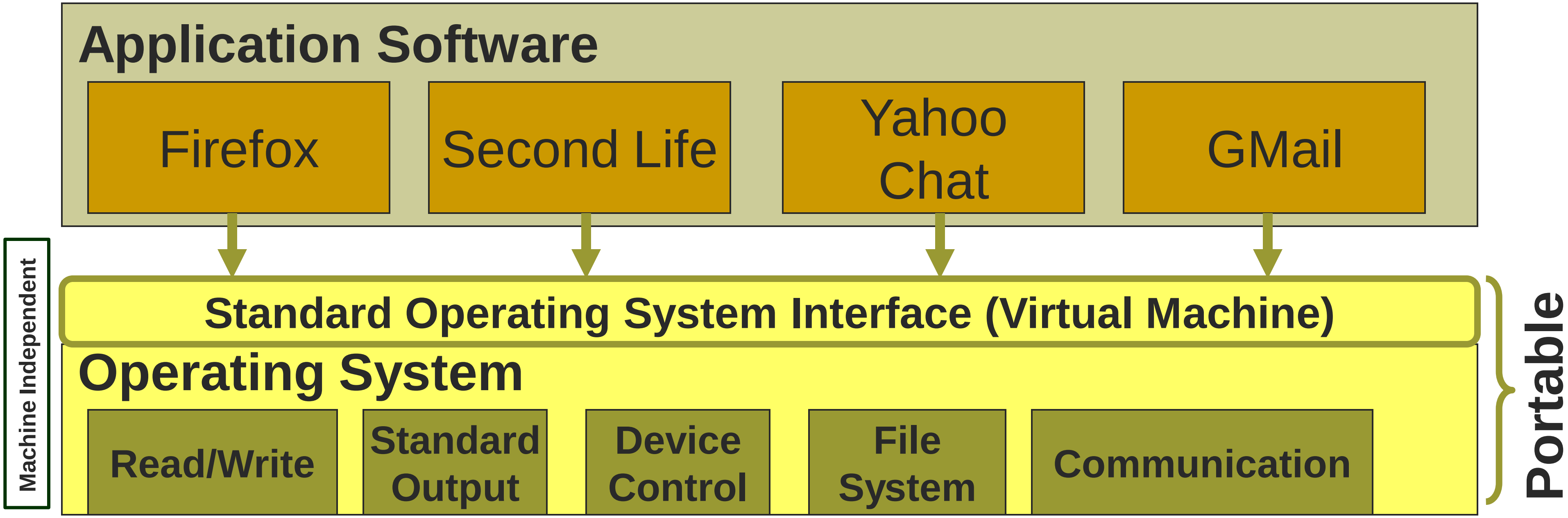
Hardware

Network

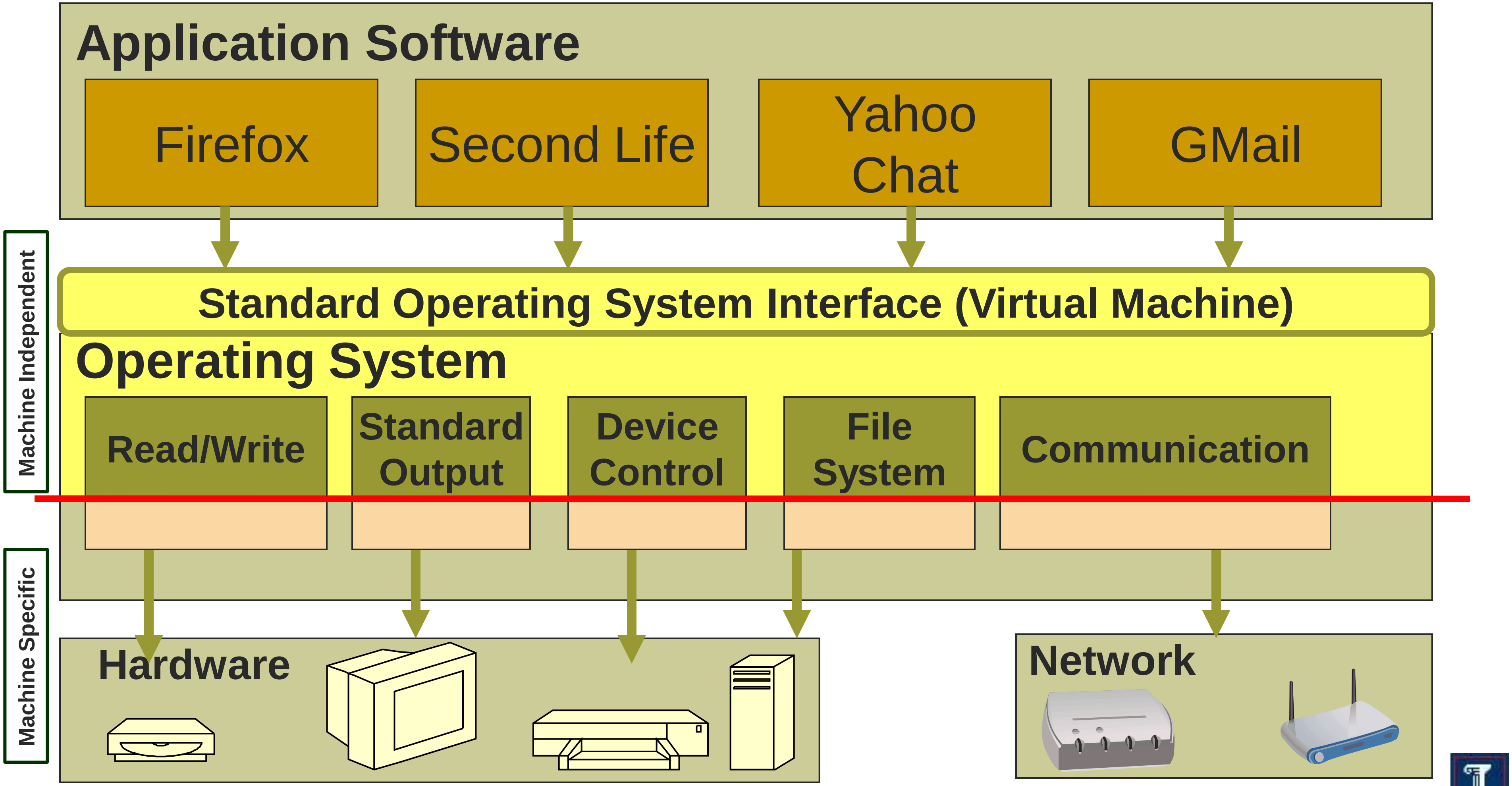
Goal: Increase Portability



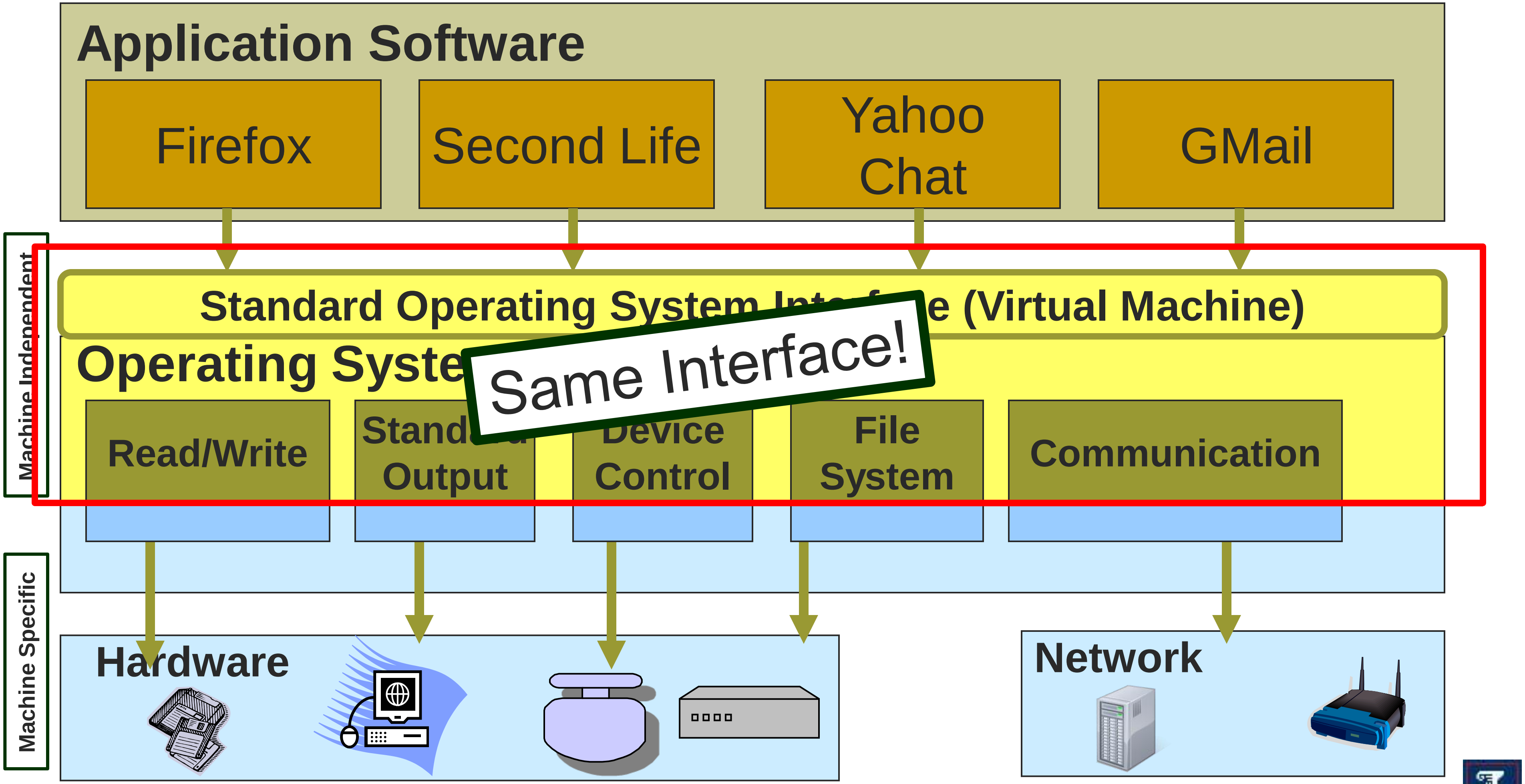
Machine Independent = Portable



OS Runs on Multiple Platforms



OS Runs on Multiple Platforms



Application Software

Firefox

Second Life

Yahoo Chat

GMail

Standard Operating System Interface (Virtual Machine)

Same Interface!

Operating System

Read/Write

Standard Output

Device Control

File System

Communication

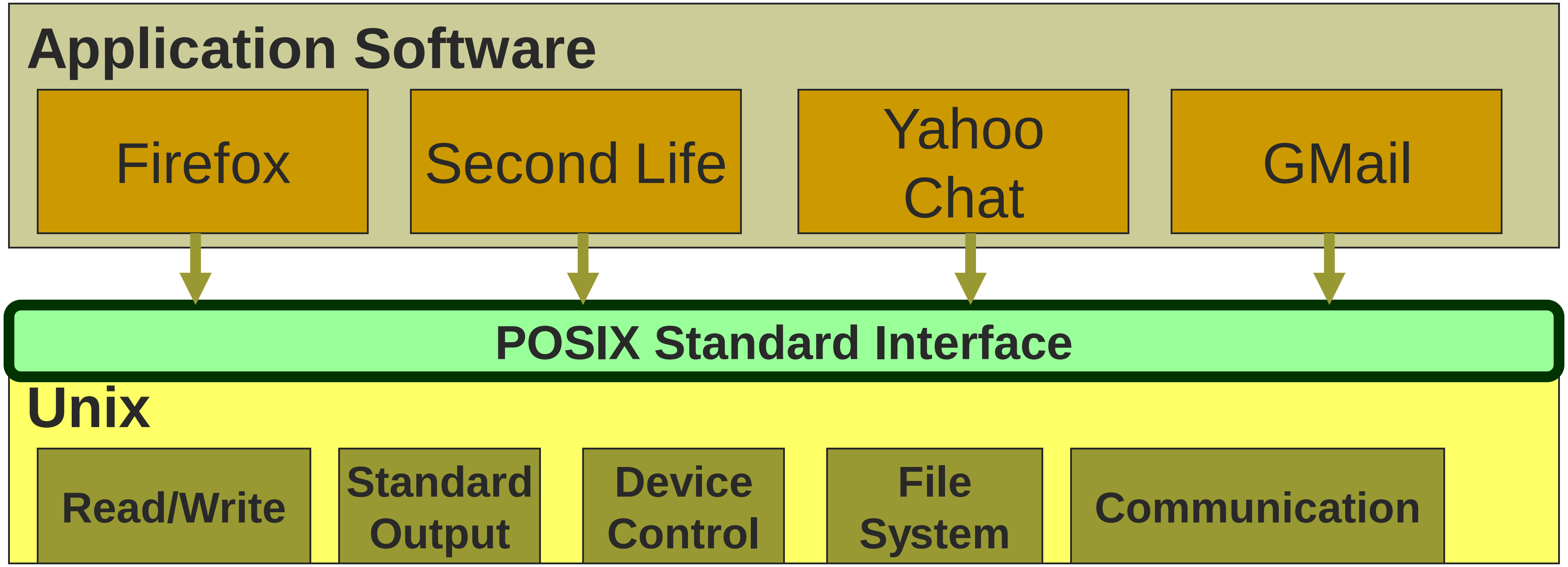
Hardware

Network

Machine Independent

Machine Specific

POSIX
The UNIX Interface Standard


Application Software
Firefox
Second Life
Yahoo Chat
GMail
POSIX Standard Interface
Unix
Read/Write
Standard Output
Device Control
File System
Communication