Advanced Operating Systems and Virtualization

Alessandro Pellegrini

A.Y. 2019/2020

Basic Information

- Lecture Schedule:
  - Course begins today! ☺
  - Lecture slots:
    - Wednesday, 17.00–19.00 (Room A3, Via Ariosto);
    - Friday, 08.00–11.00 (Room A3, Via Ariosto).
- Office Hours:
  - See on my webpage for the schedule
- Contact: pellegrini@diag.uniroma1.it

Exam Rules

- A written test (3/5 of the final mark)
- A code project (2/5 of the final mark)
  - Implementation of facilities within the Linux Kernel
  - Instructions will be given during the course

- We will see internals from Linux 2.4/2.6/3.0/4.0/5.0
  - Pick your preferred (recent) version!
  - Best if you are compatible with more than one!

Course Outline

- A Primer on Modern Hardware Architectures
- x86 Initial Boot Sequence.
- Linux Kernel Boot
- Memory Management.
- System Calls Management
- Interrupt Management
- Building the Kernel
- Kernel Data Structures

Course Outline

- Virtual File System and Devices

- Userspace Initialization

- Process Startup and Management

- Scheduling Processes

- Loadable Kernel Modules

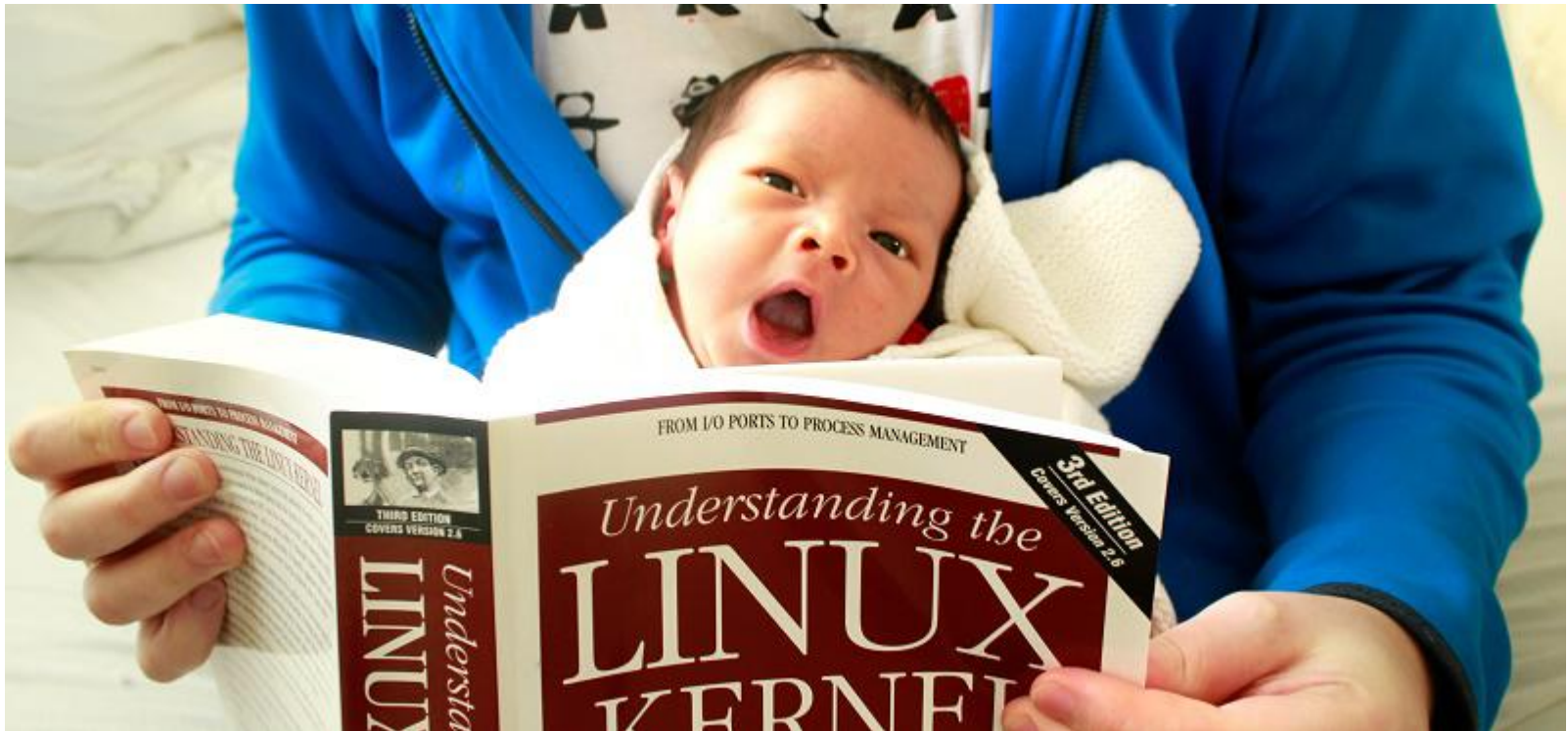- Kernel Messaging

- Security Aspects

- Hot Patching

Reference Material

- Daniel P. Bovet, Marco Cesati, *Understanding the Linux Kernel*. O'Reilly.

- Mel Gorman, *Understanding the Linux Virtual Memory Manager*. Prentice Hall.

- Alessandro Rubini, Jonathan Corbet, *Linux Device Drivers*, O'Reilly.

- David A. Rusling, *The Linux Kernel*.
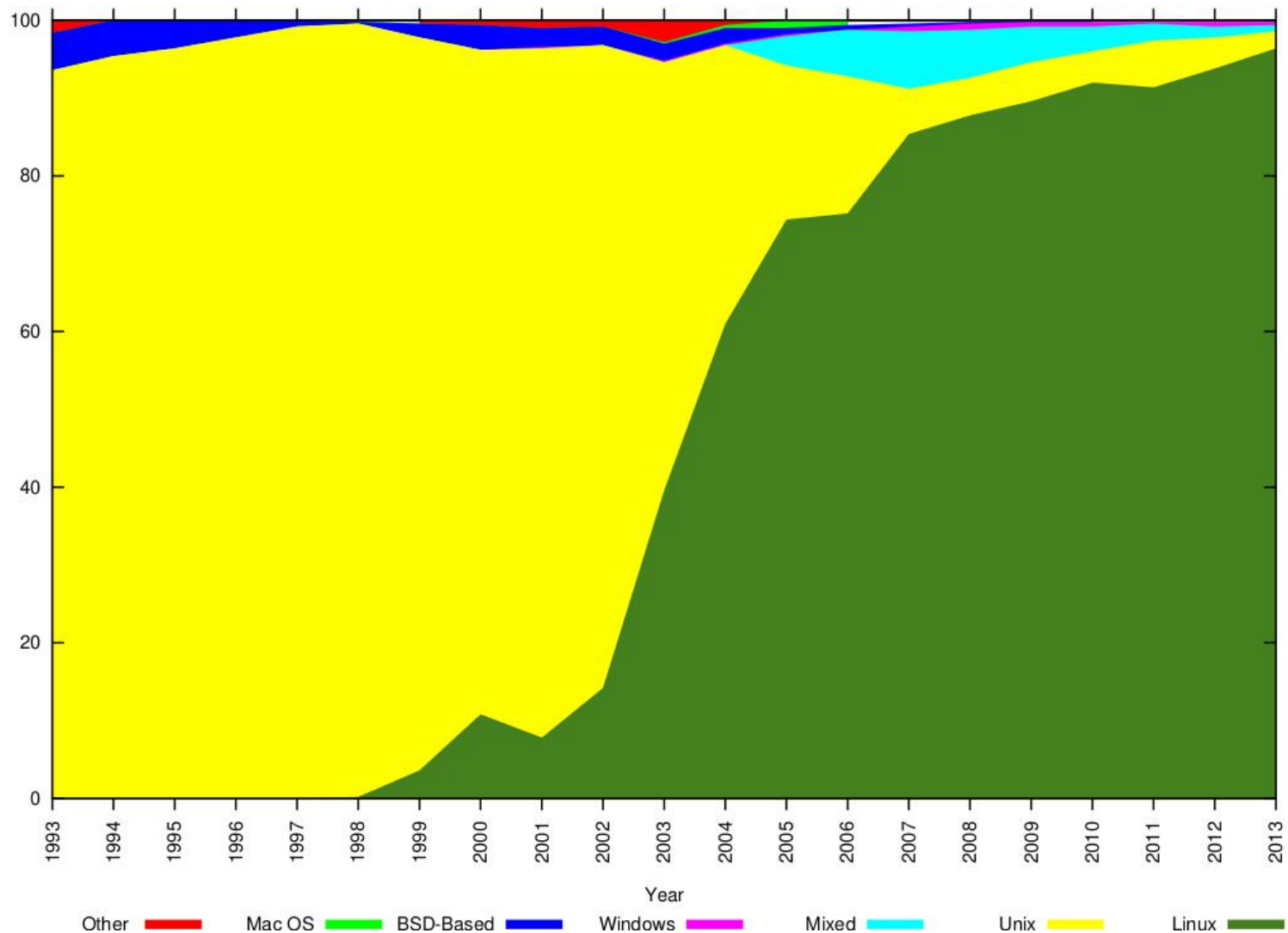
Reference Material

What you should know already

- Computing Architectures
  - Registers, I/O, Interrupts principles, flat memory model, ...
  - Numerical Representations
- Basic x86 assembly notation
- Operating Systems Principles
  - Threads and Processes
  - System Calls
- Algorithms and Data Structures
- Some notions on Concurrency
  - Synchronization, race conditions, critical sections, locks, ...
- Basic knowledge of the C programming language, and how to use a compiler

Why Linux?

Why x86?