# UNIX introduction

*Tamás Mészáros*
http://www.mit.bme.hu/~meszaros/

*Department of Measurement and Information Systems*
*Budapest University of Technology and Economics*

# Overview

- This lecture...
  - introduction, a brief history, market trends
  - overview of a UNIX kernel
  - UNIX distributions (you can try some of them at home)
  - end user perspective
  - administration point of view

- Later in this semester...
  - processes and their life cycle
  - inter-process communication
  - virtual and distributed filesystems

- Homework in virtual machines
  - **Try them out!**

# Get familiar with the UNIX operating system!

- There will be some very basic exercises you can solve
  - Demos presented during the lectures
  - Practice basic commands
  - Get familiar with the graphical user environment
  - Sometimes there will be a small, optional homework

- How to perform these exercises
  - The easiest way is to use a virtual machine
  - Download the free Vmware Player (www.vmware.com/player)
  - Download UNIX images from http://www.thoughtpolice.co.uk/vmware/
  - You can also create your own virtual machine (it's easy!)
  - For servers: CentOS 5
  - For desktop: Ubuntu variants
    - Ubuntu (Gnome3 + Unity)
    - Ubuntu Gnome Remix (Gnome3 + Gnome Shell)
    - Kubuntu (KDE)
    - Xubuntu (XFCE)

# Advanced exercises

- Typical administrative tasks (security setup, system tuning, etc.)
- Application install and setup

```
01-software-repos.setup
02-basic.setup
03-bash.setup
04-time.setup
05-security.setup
06-logging.setup
10-authentication.setup
11-ssh-login.setup
15-ldap-server.setup
16-antivirus.setup
21-storage.setup
23-iscsi.setup
24-quota.setup
25-autofs.setup
27-nfs.setup
28-samba_cifs.setup
29-inotify.setup
31-database.setup
43-bacula.setup
52-smtp.setup
55-dovecot.setup
61-swdevel.setup
71-webserver.setup
73-php.setup
76-owncloud.setup
77-horde-webmail.setup
79-maillist.setup
81-dns-server.setup
82-dhcp-server.setup
85-shibboleth.setup
91-monitoring.setup
95-kernel-tuning.setup
97-vmguest.setup
98-update.setup
99-firewall.setup
```

# Why we talk about UNIX?

- 40+ years of development and application
  - hundreds of different hardware platforms
  - hardware and software evolved in parallel
  - cradle for many today's technologies
- the source code was (and most of the time is) available to study
  - (not just Linux)
- many very different areas of applications
  - from mobile phones to hundred-core servers
- „small is beautiful” (at least in the beginning)
  - simple principles
  - common solutions to different problems
  - reusable ideas, architectures and program code

# In the beginning there were....

The Accelerated Strategic Computing Initiative Program includes participants from Los Alamos National Laboratory, Lawrence Livermore National Laboratory, and Sandia National Laboratories.

LALP-98-127

# A brief history of how UNIX born

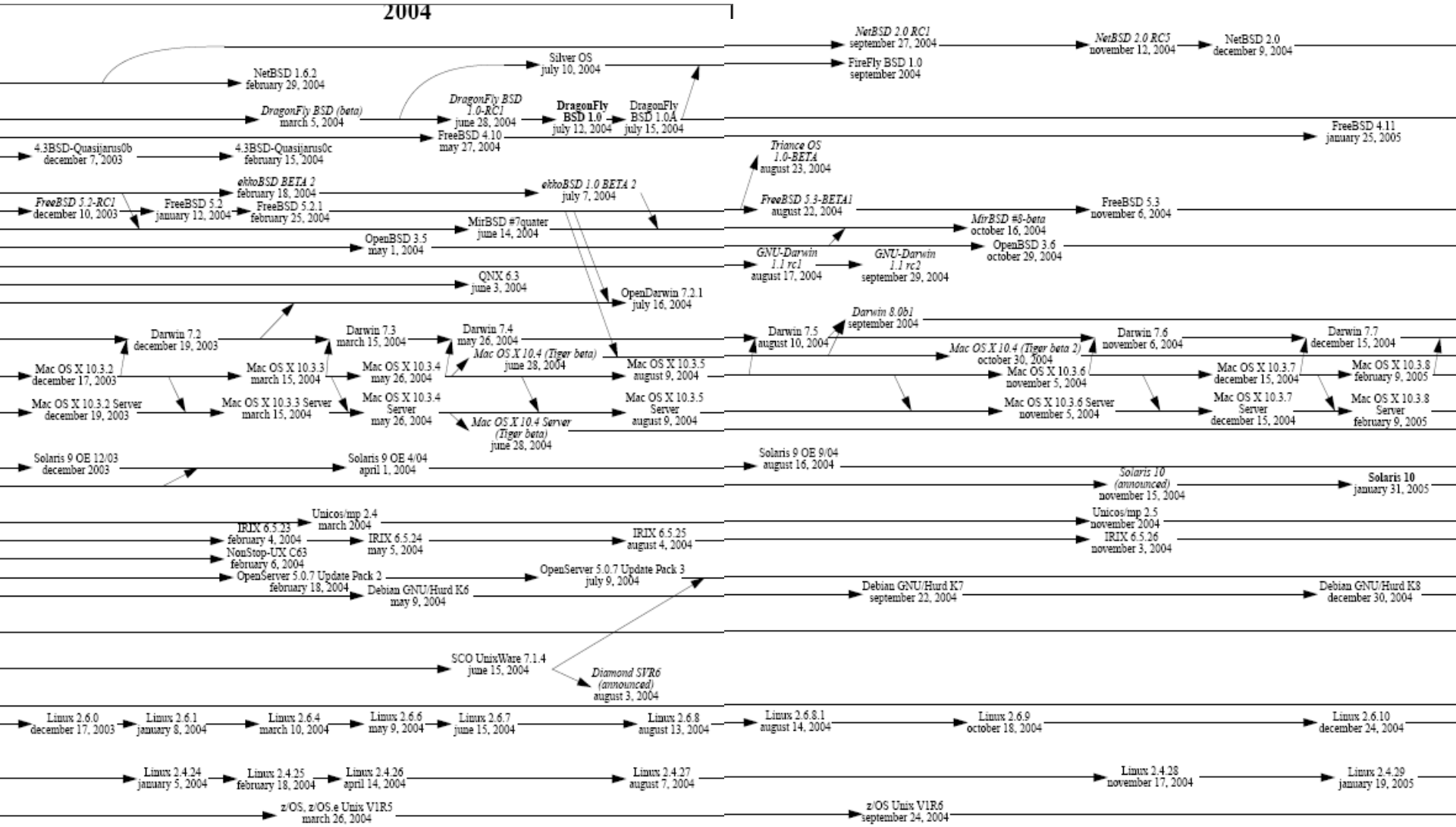
- AT&T Bell Lab, 1969, Ken Thompson, Dennis Ritchie
  - Space Travel game, expensive processor time, a spare PDP-7
  - too sloooow execution system for their game
  - no suitable programming language to implement their ideas
  - so the created an operating system (later named UNIX by Brian Kernighan)
  - and a programming language (called C)

- US antitrust law affected their employer (AT&T)
  - could not make business form the new operating system
  - academic institutions were given a copy of the source code

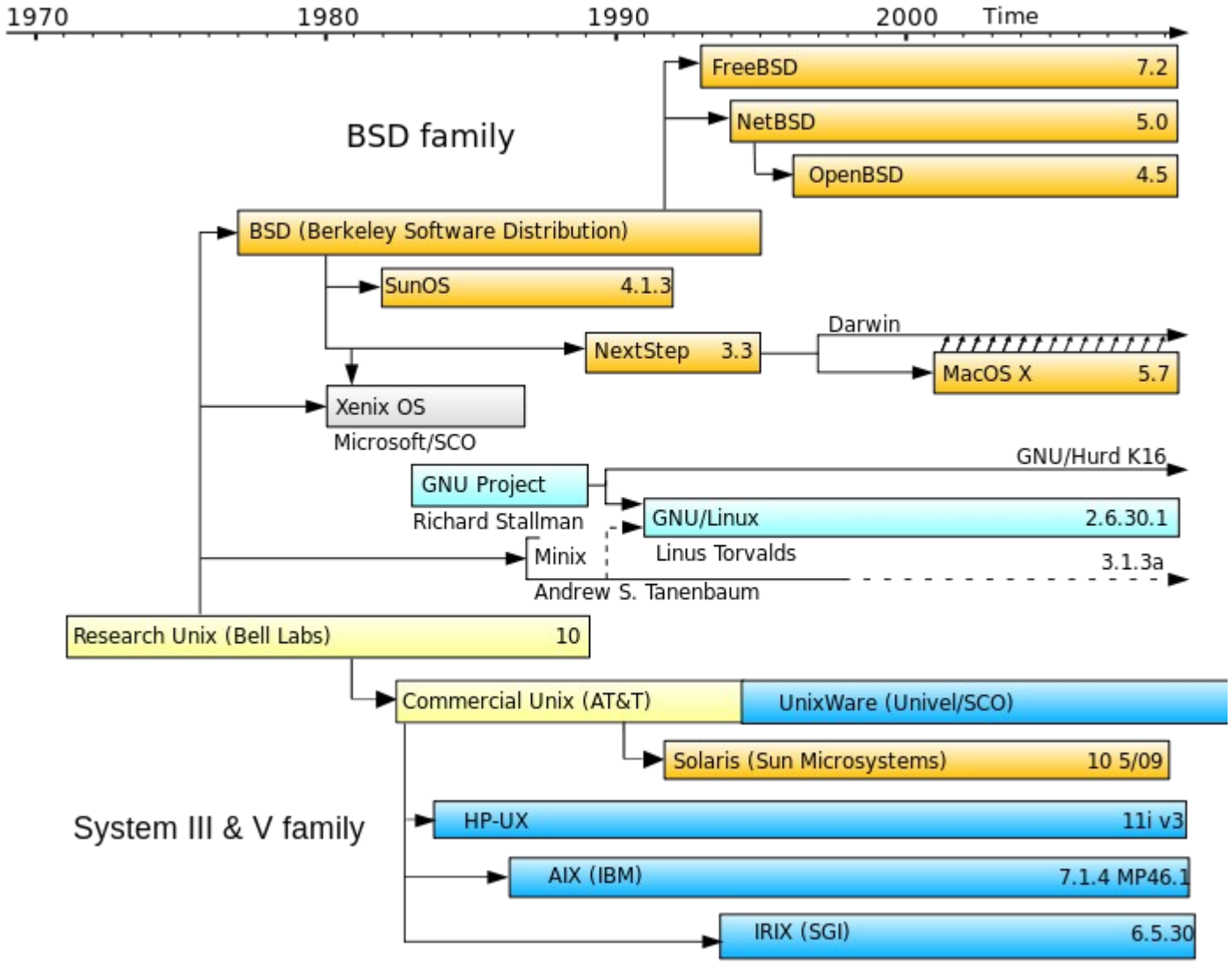
- Great interest in UNIX (and C) by research and academic people
  - 1976-77: Ken Thompson gave a lecture at Univ. California-Berkeley

- See http://www.levenez.com/unix/ for more background info

# UNIX family tree




Forrás: http://www.levenez.com/unix/ (2005)




Forrás: http://www.theopensourcery.com (2011)

# UNIX and standardization

- This family is a bit chaotic – let's make a common ground
- Standardization: *de jure* and *de facto* standards
- Main UNIX (de facto standard) branches
  - System V
    - AT&T: Sun Solaris, SCO, ...
  - BSD
    - Berkeley: SunOS, OpenBSD, FreeBSD, ...
- de jure standardization
  - IEEE POSIX
  - Linux Standards Base (LSB)
  - AT&T SVID (pl. SVR4)
  - Open Group: X/Open, Unix95, Unix98, …

# Market trends

- Client platform (undergoes heavy changes these years)
  - PC platform:
    Windows 92%, Mac OSX 6,4%, Linux 1,6% (2012.01)
    (Pár éve: Windows 94%, Max 5%, Linux 1%)
  - PC + smart phone + tablet:
    Windows 20%, OSX/iOS 24%, Android 42% (2012.12.)

    *See the figure on the following slide!*

  See

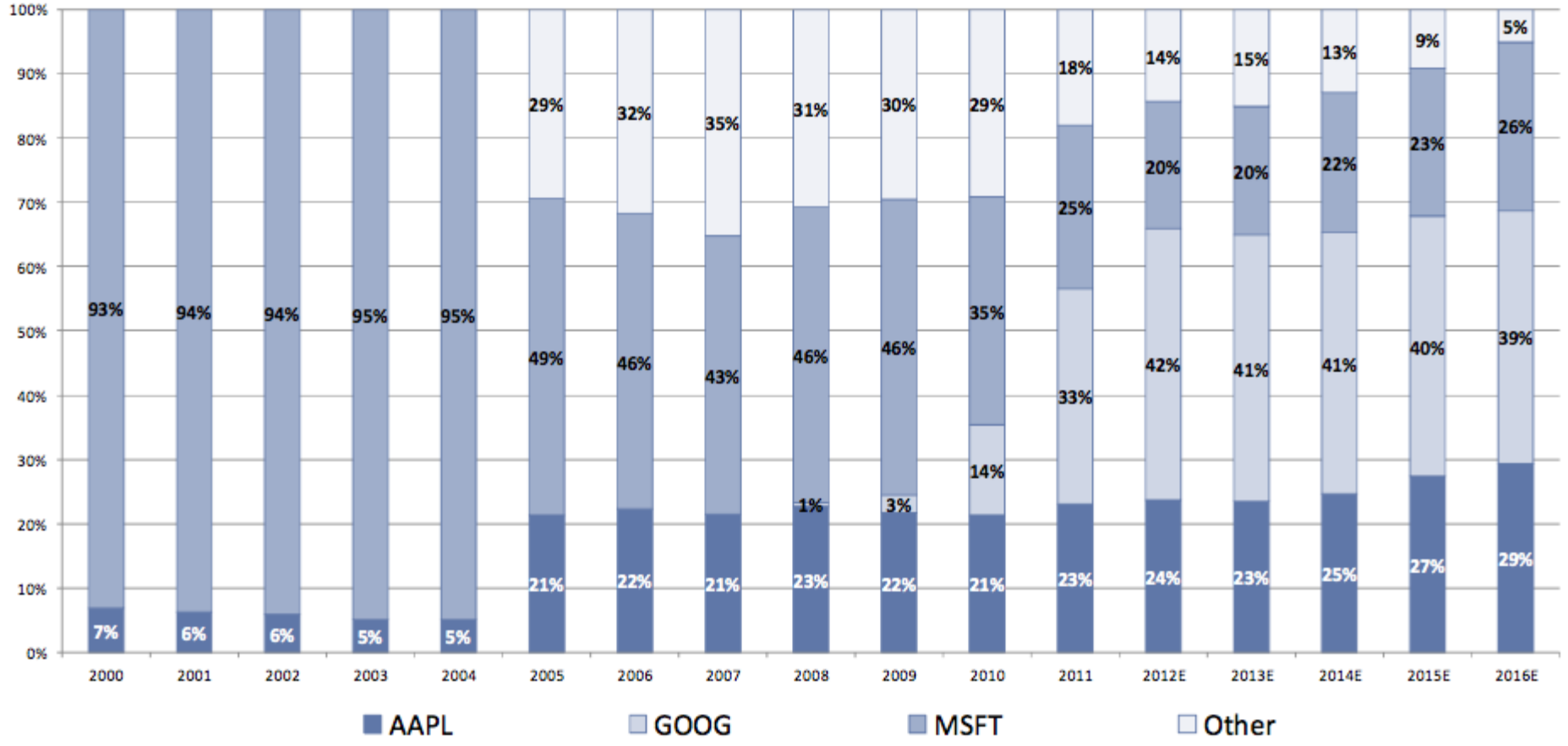
  *http://en.wikipedia.org/wiki/Usage_share_of_operating_systems*

- Server platform
  - UNIX variants 40%, growing Linux, disappearing other UNIX
  - Windows 50%

- (+ embedded systems, industrial machines, etc.)

**Exhibit 1: Vendor share of consumer compute, 2000-2016E**

Shift from single-vendor dominance (MSFT) to multiple vendors (AAPL, GOOG, MSFT, Other)




*Source: IDC, Goldman Sachs Research.*

# Main favors of UNIX

- Online catalog: http://distrowatch.com/
- Server
  - RedHat Enterprise Linux and its clones (CentOS, Scientific Linux)
  - SUSE Linux Enterprise Server / openSUSE (Novell)
  - Debian
  - Ubuntu Server
  - OpenBSD and variants (FreeBSD, NetBSD, ...)
  - Oracle (Sun) Solaris (and OpenSolaris)
- Desktop
  - Ubuntu (Kubuntu, Edubuntu, Mythbuntu, …)
  - Linux Mint
  - Arch Linux
  - Fedora (RHEL „sandbox”)
  - SUSE Linux Enterprise Desktop / openSUSE (Novell)
- Embedded (industrial, consumer, telco, etc.)
  - just an interesting example: OpenElec.tv

# The UNIX kernel

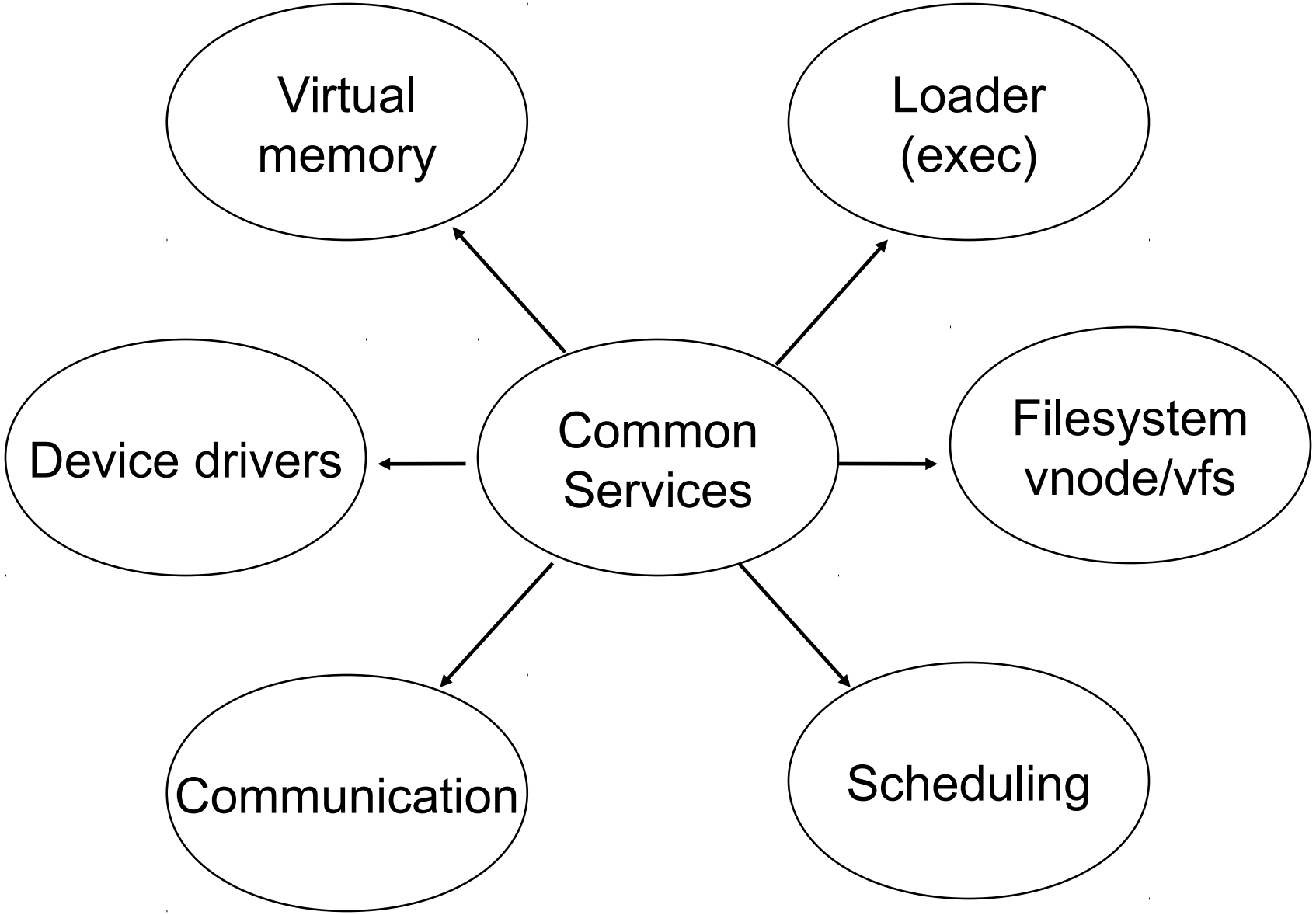
- How big is a kernel source code?
  - Linux 3.1: 37 thousand files and 14 million program lines
  - MINIX: core kernel < 1400 lines, full kernel ~ 5000 lines

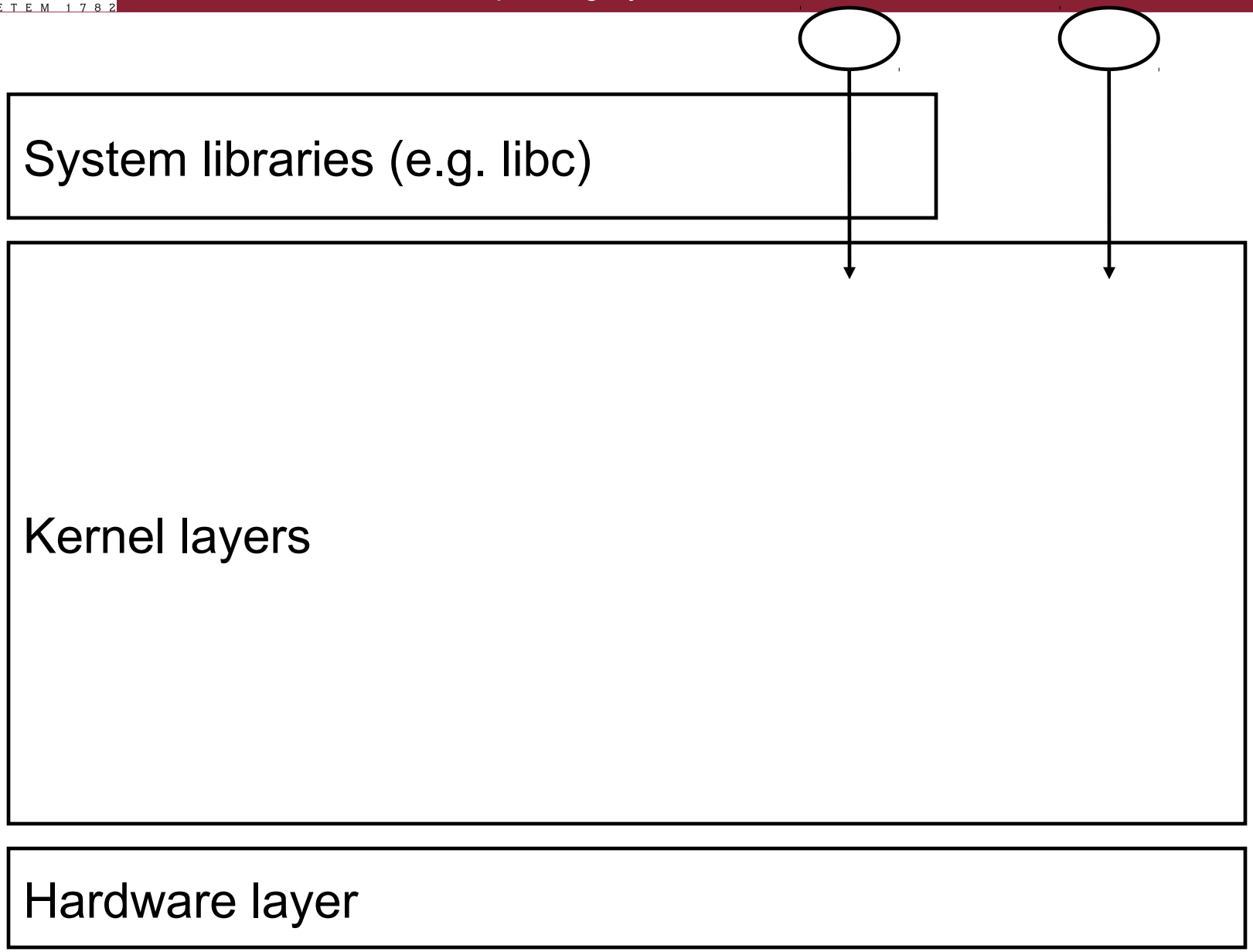
- Linux kernel 3D visualization
  http://www.pabr.org/kernel3d/kernel3d.html
  http://blog.mit.bme.hu/meszaros/node/164

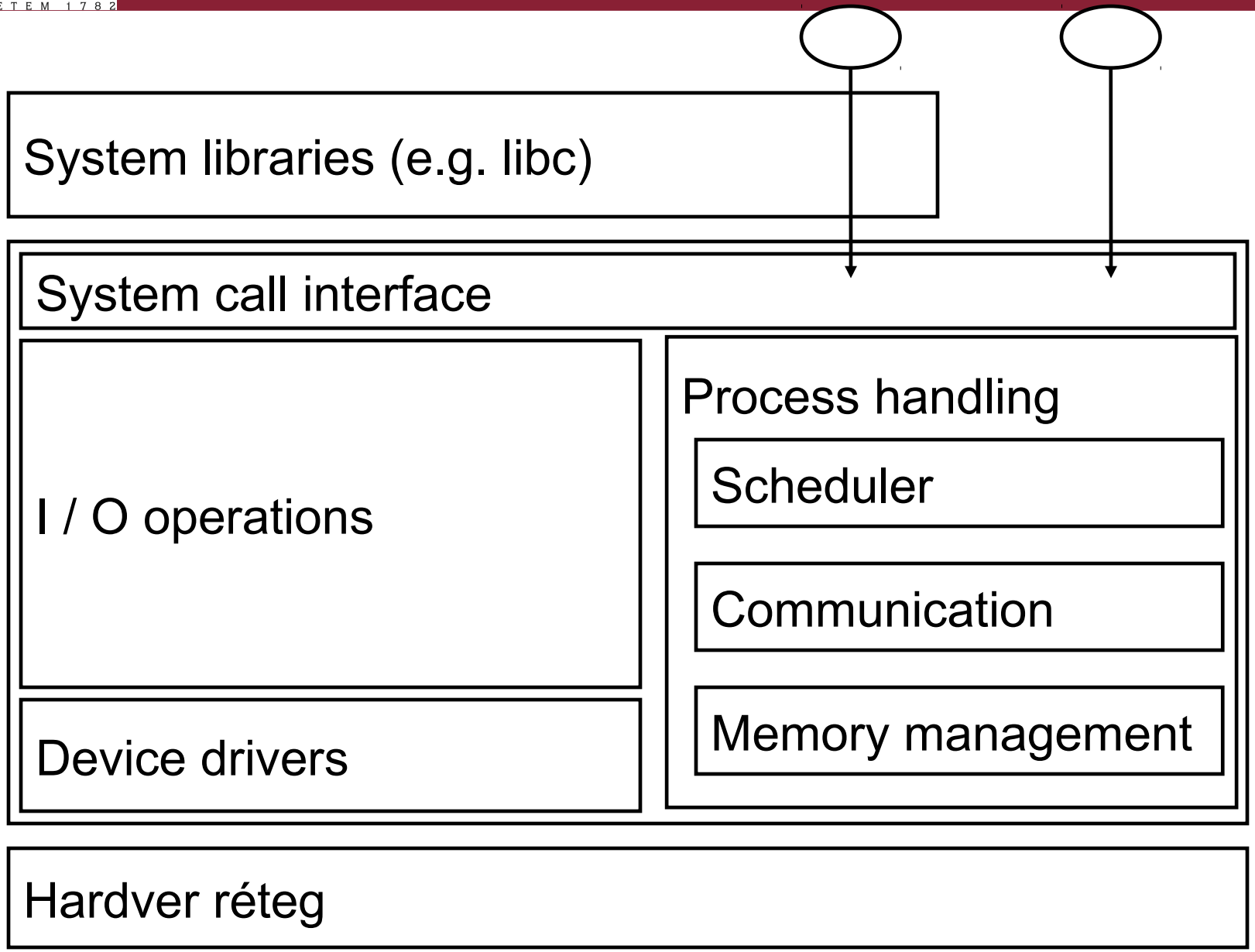
- More to read:
  - http://www.jukie.net/bart/blog/linux-kernel-walkthroughs
  - http://en.wikiversity.org/wiki/Reading_the_Linux_Kernel_Sources

# Basic concepts of building a kernel

- Layered
  - interfaces, standards, open systems
- Modular
  - online changing structure, dinamic, extensible, reusable
- Microkernel
  - lot of small kernel processes + message passing mechanism
- Monolithic kernel
  - a big process with internal function calls
- UNIX has typically a monolithic, modular, layered kernel


Virtual memory
Loader (exec)
Device drivers
Common Services
Filesystem vnode/vfs
Communication
Scheduling


System libraries (e.g. libc)
Kernel layers
Hardware layer


System libraries (e.g. libc)
System call interface
I / O operations
Device drivers
Process handling
Scheduler
Communication
Memory management
Hardver réteg

# UNIX for end users

- Very appealing graphical environments
  - fully usable over the network, symmetric multi-user environment
  - a wide variety of desktop environments:
    Gnome2, Gnome Shell, Unity, KDE, XFCE, etc.

- The directory hierarchy is a bit different than in Windows
  (No drives...)

- A large set of (traditional) command line programs
  - manuals: man, info, who (am I) és w
  - file operations: ls, cp, mv, more és less, vi, zip, tar
  - process operations: ps, kill, at, nice, >, >>, <, &

- All typical desktop applications are available on UNIX
  - network: web, email, chat, …
  - office: document editing (no MS word though), calendar, workflow, ...
  - programming environments and languages including C# (mono)

# UNIX for sysadmins

- Character terminal config (preferred) and graphical utilities
- User management and access control
  - basic identification: user – group – others and related access rights
  - advanced: selinux, RBAC (role-based control)
- Logging, monitoring and accounting ($$$$)
- Hardware drivers
  - many very different hardware platform (not just PC) and peripheral dev.
- Networking
  - Communication, distribution, load balancing
- and many more aspects

# Demonstrations

# (optional) home work #1

- Install (or download) **two** of the following UNIX systems!
  - **CentOS** (RedHat Enterprise Linux clone, **necessary for demos**)
  - **Ubuntu** (most popular desktop Linux)
  - **Ubuntu Gnome** (same with a different GUI)
  - **Xubuntu** (yet another graphical interface)

- Compare the following graphical user interfaces!
  - CentOS: **Gnome2** („classic” Linux)
  - Ubuntu: **Unity** (a Gnome3 variant)
  - Ubuntu Gnome: **Gnome3 + Gnome Shell**
  - Xubuntu: **XFCE** (a lightweight windowing environment)

- More tips on server side: http://blog.mit.bme.hu/meszaros/rhsetup

- Tips for Gnome3 desktop: http://blog.mit.bme.hu/meszaros/gnome3

# Summary

- UNIX has a very divers world:
  - in different environments, on different hardware platforms for different purposes
- Simple underlying concepts
  - portability, compatibility, cooperation, openness
- During this semester
  - get familiar with UNIX (desktop, maybe server side)
  - get acquainted with some basic areas (processes, communication, ...)
  - (optionally) perform some exercises in a virtual machine
- If you dare
  - try a UNIX desktop environment
  - try to install and setup some common services (e.g. Web, mailing, ...)

# (+1) When do the World end?

When will be the end of UNIX time?

The exact time is a 32bit signed integer on UNIX counting seconds.

It started on January 1. 1970. The actual time is:
http://www.coolepochcountdown.com/

January 19. 2038. will mark the end of our time when it turns into a negative number (32bit systems, overflow).

On 64bit machines this time is “slightly” different
December 4, 292 277 026 596.

See:
http://www.youtube.com/watch?v=QJQ691PTKsA
http://en.wikipedia.org/wiki/Year_2038_problem