



ΕΠΛ421 - Προγραμματισμός Συστημάτων

Διάλεξη 1

Εισαγωγή

Δημήτρης Ζεϊναλιπούρ



Επαναληπτικά

- **Λογισμικό Εφαρμογών (Application Software):**
 - Περιλαμβάνει προγράμματα που επιτρέπουν την **εκτέλεση συγκεκριμένων εργασιών** των χρηστών π.χ την δημιουργία κειμένων, την σχεδίαση γραφικών, την οργάνωση δεδομένων κλπ.
- **Λογισμικό Συστήματος (System Software):**
 - Περιλαμβάνει το σύνολο των προγραμμάτων που ελέγχουν διαχειρίζονται και **συντονίζουν** τους **πόρους των Η/Υ**.
 - Λειτουργεί σε **καθεστώς ανεξαρτησίας** από συγκεκριμένες εφαρμογές.
 - Η λειτουργία του **ΔΕΝ** είναι **άμεσα αντιληπτή** από τον απλό χρήστη.
- **Προγραμματισμός Συστημάτων;**
 - Η διαδικασία συγγραφής λογισμικού συστήματος.

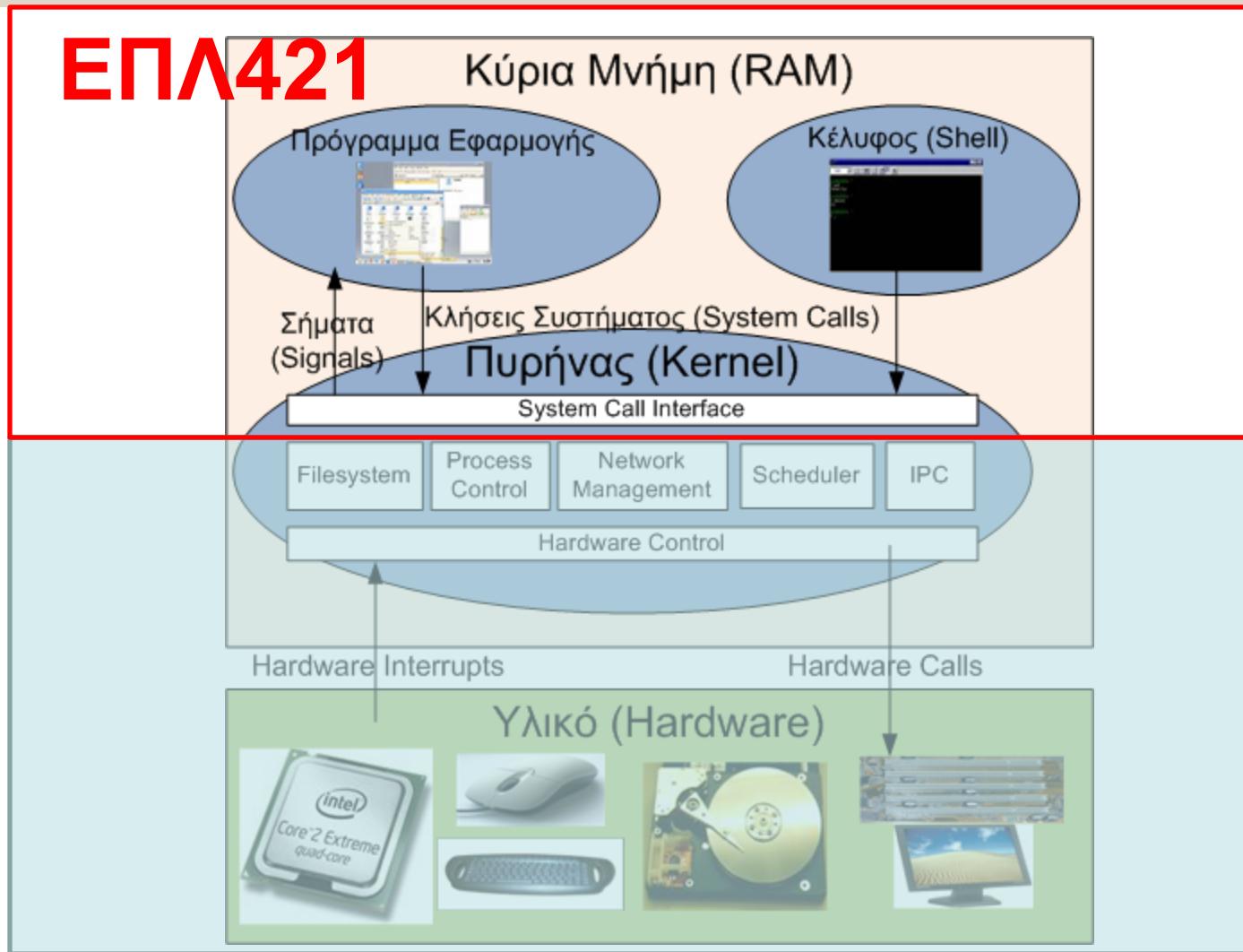


Τι καλύπτει το ΕΠΛ421;

Βασικές και Προχωρημένες έννοιες Προγραμματισμού Συστημάτων

Στόχος: Να κατανοήσουν οι φοιτητές/τριες τους μηχανισμούς με τους οποίους επιτυγχάνεται η πρόσβαση στις ρουτίνες (βιβλιοθήκες / μονάδες) ενός Λειτουργικού Συστήματος (UNIX) με τη χρήση της γλώσσας προγραμματισμού C, δημοφιλών κελυφών και εργαλείων ωφελιμότητας.

Τι καλύπτει το ΕΠΛ421;





Αναλυτικοί Στόχοι Μαθήματος

- **Εκμάθηση βασικών και προχωρημένων τεχνικών διαχείρισης και προγραμματισμού στο Λειτουργικό Σύστημα Unix (Shell commands and Bash Shell programming).**
- **Εκμάθηση προχωρημένων τεχνικών προγραμματισμού στην C & Unix:**
 1. Συστήματα Αρχείων και Διαχείριση Μνήμης;
 2. Προγραμματισμός Διεργασιών (Processes – Πολυδιεργασιακός Προγραμματισμός) και Νημάτων (Threads – Πολυνηματικός Προγραμματισμός);
 3. Επικοινωνία Διεργασιών (Διαδιεργασιακή Επικοινωνία);
 4. Προγραμματισμός Δικτύου (Socket programming);
 5. Ασφάλεια Λογισμικού Συστημάτων (security)



Συμβόλαιο Μαθήματος

- **Επίπεδο:** Προπτυχιακό
 - Υποχρεωτικό Μάθημα: Κατεύθυνση Υπολογιστικών Συστημάτων και Δικτύων / Τεχνολογίας Λογισμικού
- **Πίστωση:** 7.5 μονάδες ECTS
- **Προαπαιτούμενα:**
 - ΕΠΛ222: Λειτουργικά Συστήματα
 - ΕΠΛ232: Προγραμματιστικές Τεχνικές και Εργαλεία
 - ΕΠΛ221-Οργάνωση Υπολογιστών και Συμβολικός Προγραμματισμός
- **Μέθοδοι Διδασκαλίας**
 - Διαλέξεις (3 ώρες εβδομαδιαίως)
 - Φροντιστήριο (1 ώρα εβδομαδιαίως)
 - Εργαστήριο (2 ώρες εβδομαδιαίως)



Συμβόλαιο Μαθήματος

- **Αξιολόγηση**

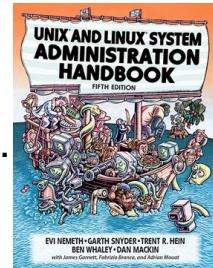
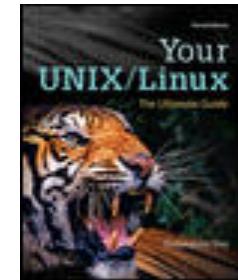
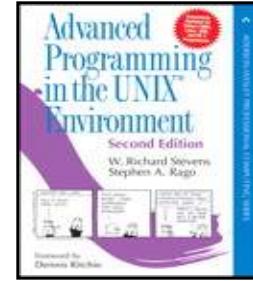
- 50% Τελική Εξέταση
- 25% Ενδιάμεση Εξέταση-**Παρ. 25/10 (8η Εβδ.)**
- 25% Ασκήσεις
 - 10% **Προγραμματιστικές Ασκήσεις**
 - 10% Project – **Υλοποίηση Συστήματος**
 - 5% Παρουσίαση κάποιας **γλώσσας** ή **βιβλιοθήκης**, με έμφαση στο προγραμματισμό συστημάτων (π.χ., Systems programming in windows, scripting languages, emerging systems in mobile, clouds, etc.)



Βιβλιογραφία

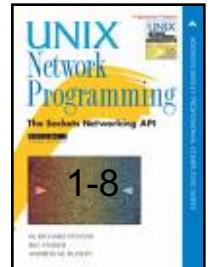
Main Bibliography

- **Stevens & Rago** [Advanced Programming in the UNIX® Environment](#), Third Edition, Richard Stevens, Stephen A. Rago, Addison-Wesley, ISBN-10: 0321637739 | ISBN-13: 978-032163773, **2013**.
- **Das** [Your UNIX/Linux: The Ultimate Guide](#), 3rd Edition, Sumitabha Das, McGraw Hill, ISBN-13 9780073376202, 800 page, **2013**.
- **Das** [UNIX and Linux System Administration Handbook](#) 5th Edition, Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley, Dan Mackin, 1232 pages, Addison-Wesley, ISBN-10: 0134277554 | ISBN-13: 978-01342775545, **2017**.



Supplementary Bibliography

- **CS:APP3e** [Computer Systems: A Programmer's Perspective, 3/E \(CS:APP3e\)](#), Third Edition, Randal E. Bryant and David R. O'Hallaron, Carnegie Mellon University, ISBN-13: 978-0134092669 | ISBN-10: 013409266X, 1128 pages, **2015**.
- **Robbins & Robbins** [UNIX Systems Programming: Communication, Concurrency and Threads](#), 2nd Edition, Kay Robbins and Steve Robbins, Prentice Hall, ISBN-10: 0134424077, ISBN-13: 978-0134424071, **2015**.
- **Stevens, Fenner & Rudoff** [Unix Network Programming, Volume 1: The Sockets Networking API](#), 3rd Edition, W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Addison Wesley Professional, ISBN-10:0131411551 | ISBN-13:9780131411555, **2004**.





WWW

- Όλες οι πληροφορίες σχετικά με το μάθημα βρίσκονται στο ακόλουθο URL

<http://www.cs.ucy.ac.cy/courses/EPL421>

A screenshot of a web browser showing the University of Cyprus course page for EPL421: Systems Programming. The page includes the university logo, navigation links for EPL421, News, Schedule, Labs, Assignments, Links, Presentations, Moodle, TA, and a search bar. The main content area displays course details: Instructor (Demetris Zeinalipour), Type (Upper-division Undergraduate (Elective)), Prerequisite (EPL222: Operating Systems), When (Tue./Fri., 16:30-18:00 in ΧΩΔΟ1-#101), Recitations (Wed., 15:00-16:00 in ΧΩΔΟ1-#004), Laboratory (Fri., 18:00-20:00 in ΘΕΕΟ1-B103), and Assistant (Pavlos Antoniou).

Overview

In this course, students will learn to develop complex system-level software in the C programming language while gaining an intimate understanding of the UNIX operating system (and all OS that belong to this family, such as Linux, the BSDs, and even Mac OS X) and its programming environment. Topics covered will include the user/kernel interface, fundamental concepts of UNIX, user authentication, basic and advanced I/O, filesystems, signals, process relationships, and interprocess communication. Fundamental concepts of software development and maintenance on UNIX systems will also be covered. The students are expected to have a good working knowledge of the C programming language (EPL232) and a good working knowledge of fundamental Operating System Concepts (EPL221).



WWW

- Για τις εκπαιδευτικές δραστηριότητες του μαθήματος (υποβολή εργασιών, φόρουμ ανακοινώσεων, ερωτηματολόγια, βαθμολογίες εργασιών, κτλ) θα χρησιμοποιηθεί το Moodle. <http://moodle.cs.ucy.ac.cy/>

The screenshot shows the Moodle course page for EPL421 - Systems Programming. The page includes a navigation bar with links to Home, My courses, and EPL421. The main content area displays course details such as the title, instructor, and announcements. A sidebar on the right contains links for latest announcements, search forums, and information about the CS Colloquium Series.

EPL421 - Systems Programming

General

EPL421: Systems Programming
(ΕΠΛ421: Προγραμματισμός Συστημάτων)

Instructor: Demetris Zeinalipour

For more information on this course please visit
<https://www.cs.ucy.ac.cy/~dzeina/courses/epl421/>

Important Notices:

- Please check this section regularly for additions and changes.
- To view all resources available on this page you need to logon.

Topic 1

Assignment #1

LATEST ANNOUNCEMENTS
Add a new topic...
(No announcements have been posted yet.)

SEARCH FORUMS
Go
Advanced search

CS COLLOQUIUM SERIES @ UCY

Colloquium: Resource management in heterogeneous large scale communication systems, Dr. Marilia Curado (University of Coimbra, Portugal), Wednesday, September 11, 2019, 12:00-13:00 EET.



Ανατομία του ΛΣ

- Γνωρίζουμε ότι ένα **Λειτουργικό Σύστημα (ΛΣ)** συντονίζει τους πόρους ενός Η.Υ. (υλικό, λογισμικό, διαχείριση χρηστών, κτλ.).
- Το ΛΣ αποτελείται τα εξής συστατικά:
 1. **Kernel (Πυρήνας):** διεξάγει τις πιο σημαντικές λειτουργίες για την εύρυθμη λειτουργία του ΗΥ (χρονοδρομολόγηση διεργασιών, διαχείριση μνήμης, Δια-διεργασιακή επικοινωνία, κτλ.)
 2. **System Commands, Utilities and Libraries** (π.χ., ls, sed, awk, pthread.h, unistd.h, κτλ.) : Προγράμματα και βιβλιοθήκες τα οποία επιτρέπουν στα προγράμματα **εφαρμογών (user space)** να έχουν πρόσβαση στις ρουτίνες του **Πυρήνα (kernel space)** του Λειτουργικού Συστήματος

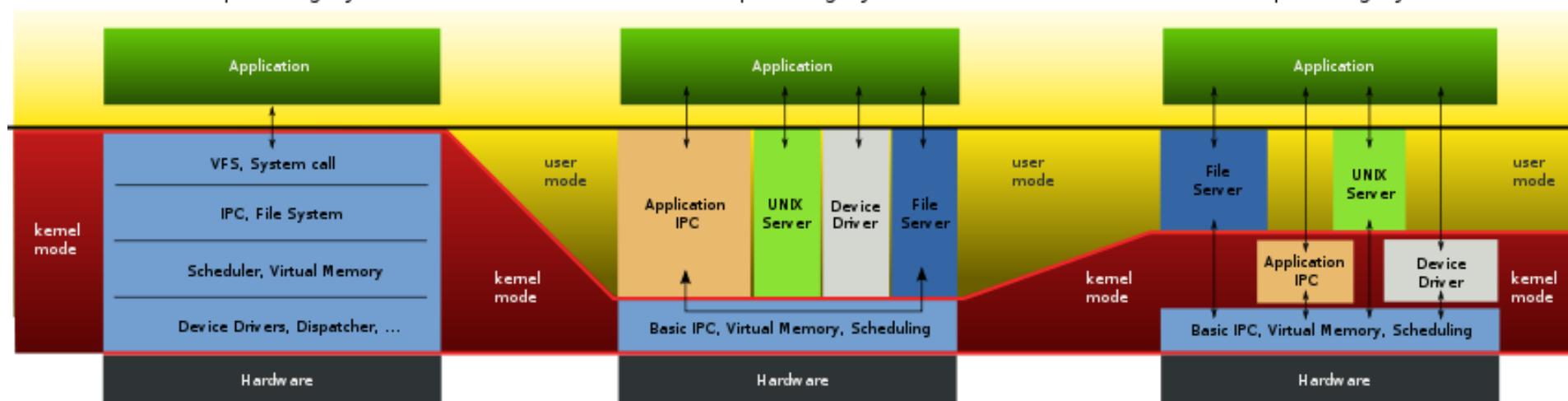


Ανατομία του ΛΣ

Monolithic Kernel
based Operating System

Microkernel
based Operating System

"Hybrid kernel"
based Operating System



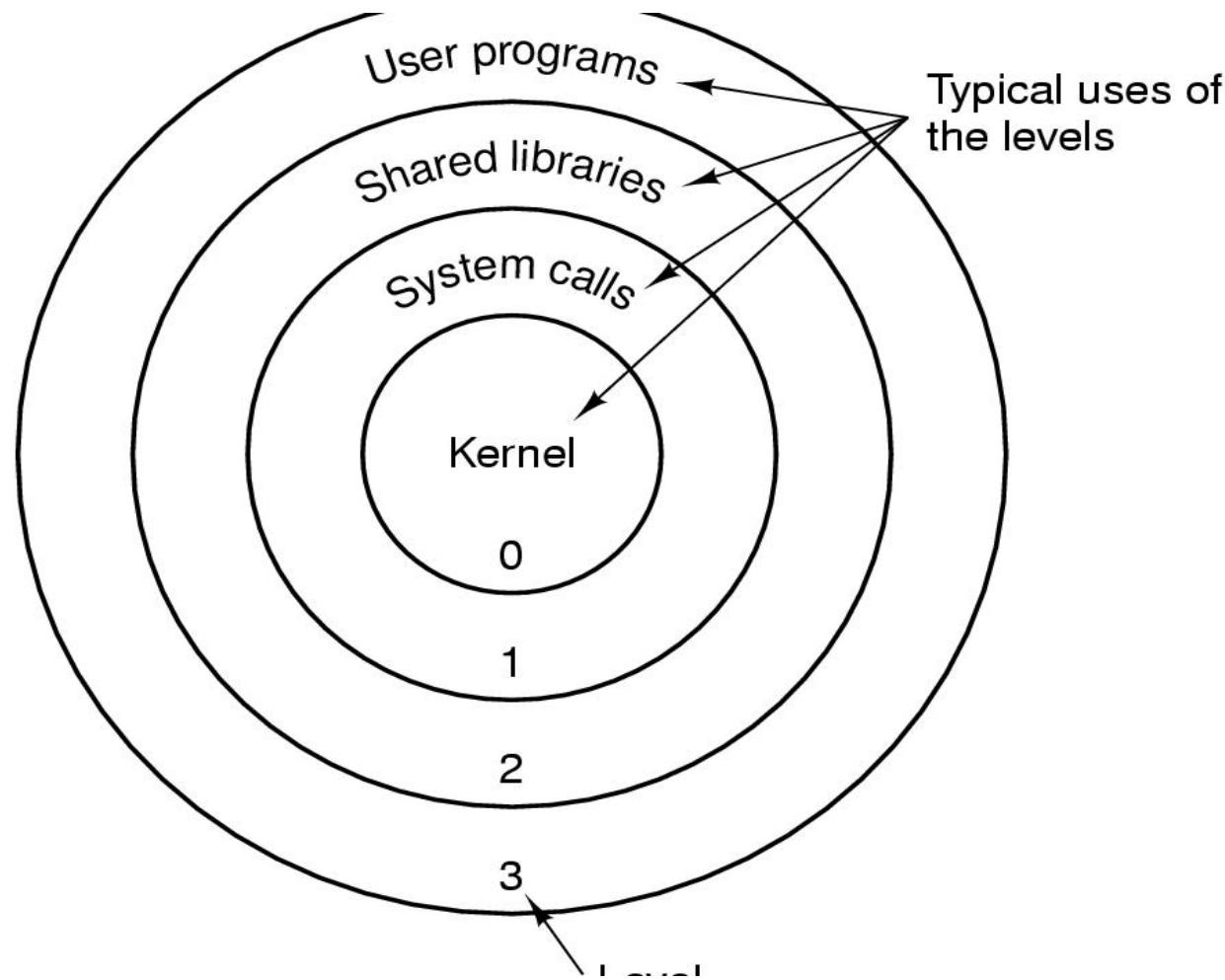
Most Modern Unixes
(Linux, CentOS, BSD-like,
Solaris, MacOS-8,
AIX, Android, κτλ)

Π.χ., Early-
UNIXes, MINIX-3
(Tanenbaum's)
read
<http://www.cs.vu.nl/~ast/reliable->

Π.χ., Mac OSX and NT-Kernel
(i.e., Leopard, Windows
NT, 2000, XP, Server
2003, Vista, Server 2008
and 7)



Πολύ-επίπεδη Πρόσβαση στους Πόρους ενός ΗΥ





Πυρήνας (Kernel)

- Το **kernel** (**πυρήνας**) είναι :
 - Ένα **πρόγραμμα** το οποίο φορτώνεται στην μνήμη ενός ΗΥ κατά την **διάρκεια της εκκίνησης** και παραμένει εκεί καθ' όλη την διάρκεια εκτέλεσης.
 - Έχει μεταξύ άλλων τα ακολούθα υποσυστήματα:
 - **Διαχείριση Διεργασιών (Process management)**
 - Schedule processes to run on CPU, Inter-process communication (IPC)
 - **Διαχείριση Μνήμης (Memory management)**
 - Virtual memory (Paging/Swapping)
 - **Διαχείριση Εισόδου/Εξόδου (I/O system)**
 - File system, Device drivers, Buffer cache



Κλήσεις Συστήματος (System Calls)

- Παρέχουν **διασύνδεση με το τον Πυρήνα**
- Στο **Linux** υπάρχουν περισσότερα από **1,000 system calls** και αυτά μπορούν κιόλας να επεκταθούν με ***Loadable Kernel Modules (LKM)***
- Αυτά χωρίζονται σε τρεις βασικές κατηγορίες
 - File/device manipulation (Επεξεργασία Αρχείων)
 - e.g. `mkdir()`, `unlink()`
 - Process control (Διαχείριση Διεργασιών)
 - e.g. `fork()`, `execve()`, `nice()`
 - Information manipulation (Διαχείριση Μέτα-πληροφοριών), e.g. `getuid()`, `time()`

Κέλυφος και Προγρ. Κελύφους / Shell and Shell Programming



- Στο ΕΠΛ421 θα δούμε πως αλληλεπιδράμε με τον πυρήνα: **α) Μέσω του Κελύφους (Bash)** και **Προγραμμάτων Ωφελιμότητας (System Utilities)** και **β) Μέσω Βιβλιοθηκών Συστημάτων (System Libraries)**

Χρήση Κέλυφους

```
who | awk '{print $1}' | sort | uniq
```

Σημείωση: Μας εκτυπώνει τα logins των χρηστών που είναι συνδεδεμένοι στο σύστημα μια δεδομένη στιγμή

- Βλέπουμε ότι με μια τέτοια γλώσσα (scripting, high level interpreted), μπορούμε να φτιάξουμε εύκολα σύνθετες λειτουργίες

who	755
awk	3,412
sort	2,614
uniq	302

7,424 γραμμές
κώδικα¹⁻¹⁶

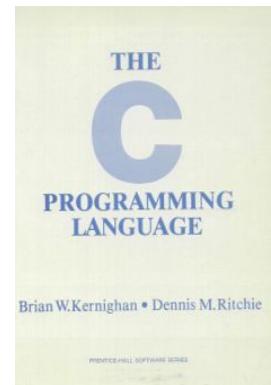


Εισαγωγή στο Unix

- **1960:** Δημιουργία του Multics (mainframe timesharing /multitasking OS by MIT, GE, Bell Labs). O Thompson εργάζεται πάνω στον υπολογιστή PDP-7 στα Bell Labs
- **1969:** Ο Ken Thomson & o Dennis Ritchie δημιουργούν ένα νέο ΛΣ για το υπολογιστή PDP-7 σε assembly το οποίο ονομάζουν Unics, το οποίο μετά ονομάζεται Unix!



- O Ritchie δημιουργεί την γλώσσα C για το Unix (**1972**) στα Bell Labs.
- ...και συγγραφέας του best-seller “The C Programming Language”!





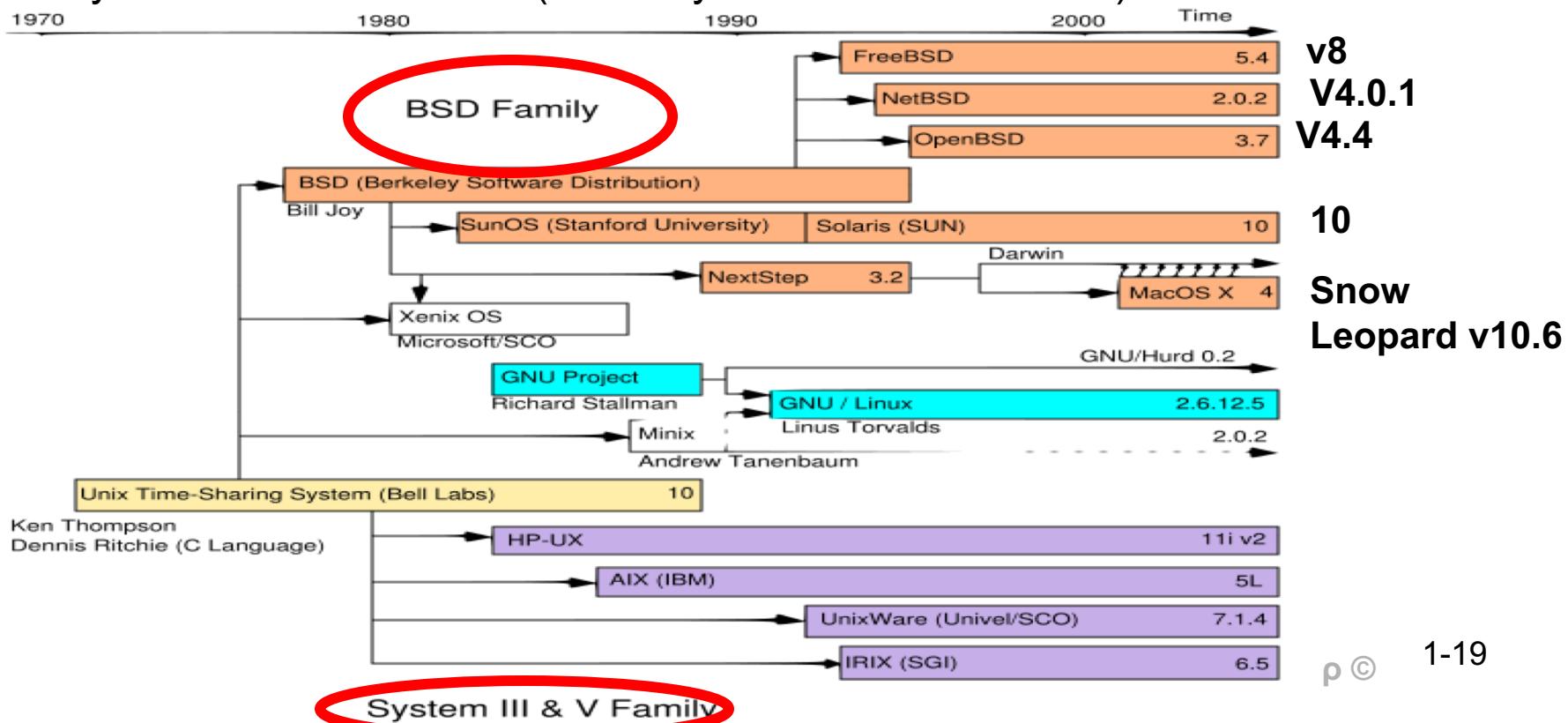
To PDP-7 (1965)





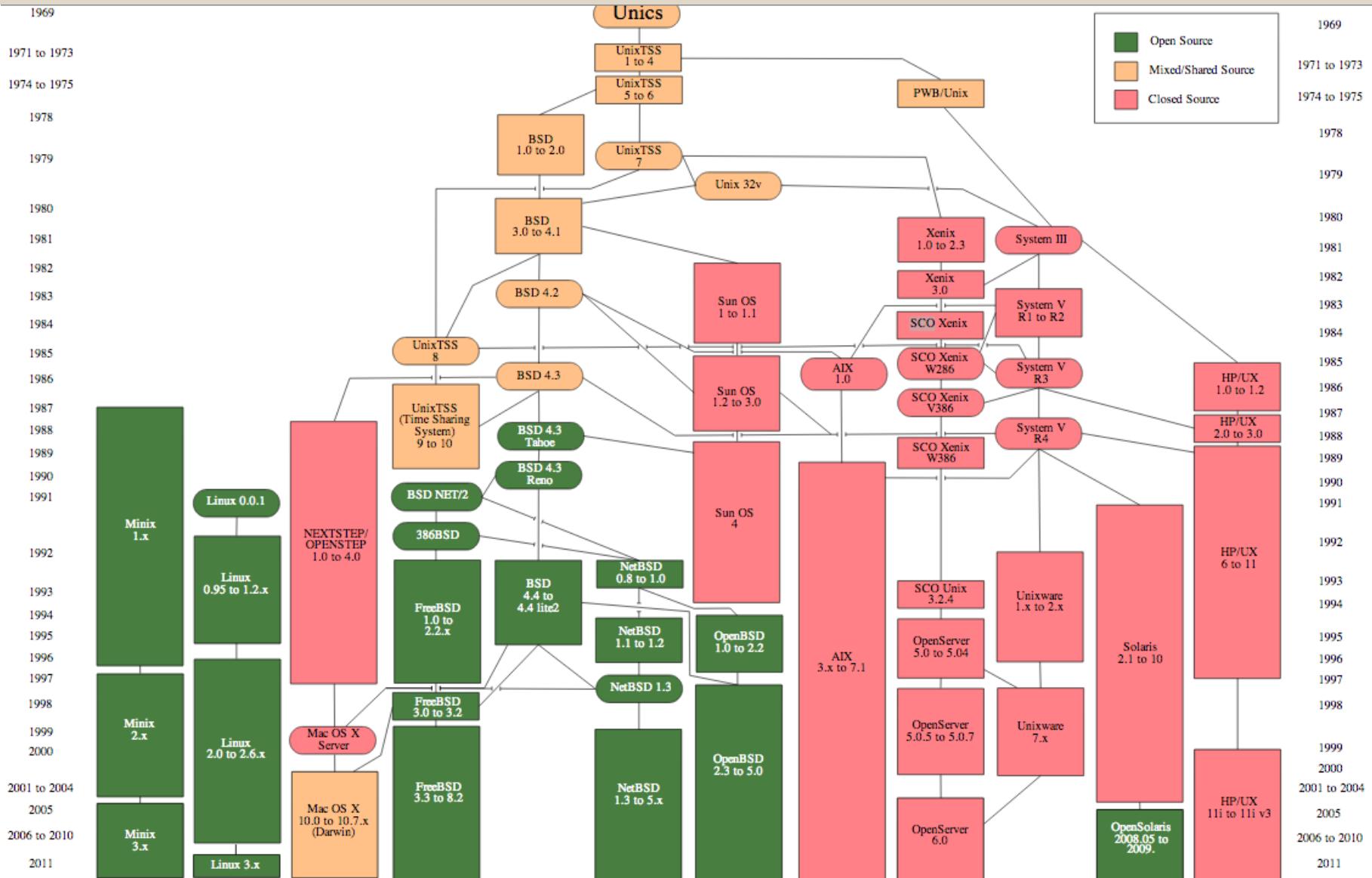
Εισαγωγή στο Unix

- 1973: Το Unix γράφεται σε γλώσσα C.
- 1976 : Το κέλυφος Bourne Shell (bsh) αναπτύσσεται στην AT&T και επιτρέπει την αλληλεπίδραση με τον πυρήνα του Λ.Σ. μέσω εντολών .
- 80's- : Το Unix συνεχίζει να εξελίσσεται. Διακρίνονται δύο οικογένειες α) Το System III&V και το BSD (Berkeley Software Distribution).





Εισαγωγή στο Unix



Εισαγωγή στο Unix

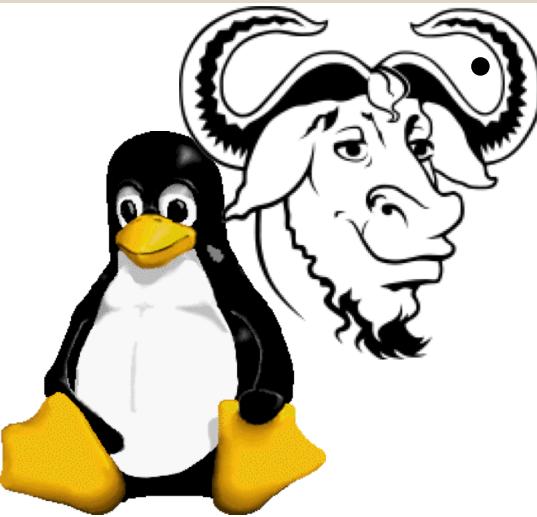
Εμπορική Επιτυχία



- AIX
- SunOS -> Solaris (Sun -> Oracle) SUS-compliant
- Ultrix, Digital Unix
- HP-UX
- Irix
- UnixWare -> Novell -> SCO -> Caldera ->SCO's Unixware
- Xenix: -> SCO
- Standardization (**IEEE POSIX**: Portable Operating System Interface for uniX (all known unices are POSIX-compliant), **SUS**: Single Unix Specification (Austin), X/open (από διάφορες ευρωπαϊκές εταιρίες))



Unix για Προσωπικούς Υπολογιστές PC



(Gnu) Linux

- Γράφεται το 1991 από τον 21 ετών Φιλανδό Linus Torvalds
- Δημιουργεί ΛΣ UNIX για PCs
- Δωρεάν με **GNU license**
- **Τελευταία Έκδοση Kernel:** 4.14 (01/2018)

• BSD Lite



- FreeBSD (1993, focus on PCs)
- NetBSD (1993, focus on portability)
- OpenBSD (1996, focus on security)
- Δωρεάν με **BSD license**



Unix για Κινητές Συσκευές

• Linux on Portable Devices

– Οι πυρήνες των πλείστων IoT συσκευών σήμερα στηρίζονται στο UNIX/Linux

- **Android** => uses and contributed to the Linux kernel (e.g., energy efficiency, drivers)
- <https://source.android.com/devices/architecture/kernel>



• **iOS (Apple)** => Darwin Kernel

- A POSIX-compliant open source OS released by Apple in 2000 (
- Darwin based on BSD -> Nextstep kernel (CMU)



• **Raspberry PI Linux**

• **Automotive Grade Linux** (e.g., VW.OS)





Unix για Υποδομές Νεφέλης (Large Filesystems - HDFS)

YAHOO!
2010

	Target	Deployed
Capacity	10PB	14PB
Nodes	10,000	4000
Clients	100,000	15,000
Files	100,000,000	60,000,000

facebook

2010

- 21 PB of storage in a single HDFS cluster
- 2000 machines
- 12 TB per machine (a few machines have 24 TB each)
- 1200 machines with 8 cores each + 800 machines with 16 cores each
- 32 GB of RAM per machine
- 15 map-reduce tasks per machine

TABLE 1: TARGETS FOR HDFS VS. ACTUALLY DEPLOYED VALUES
AS OF 2009

HDFS scalability: the limits to growth

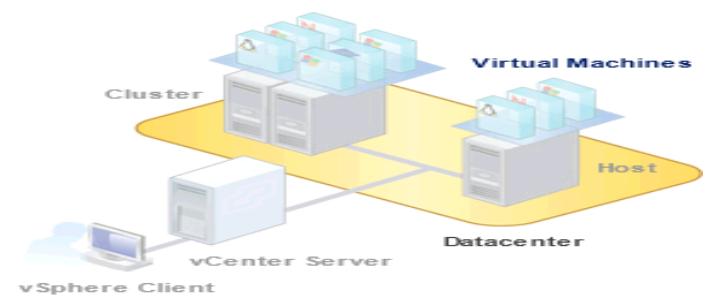
<http://static.usenix.org/publications/login/2010-04/openpdfs/shvachko.pdf>

Εικονικοποίηση (Virtualization)

- **Εικονικοποίηση (virtualization)** αναφέρεται στη διαδικασία δημιουργίας μιας νοητής, πάρα πραγματικής έκδοσης, κάποιου μέσου (π.χ., Η/Υ, μνήμης, δικτύου, υπηρεσίας, Λ.Σ., εφαρμογών, κτλ.)



Εικονικές Μηχανές



Εικονικοποίηση (Virtualization)



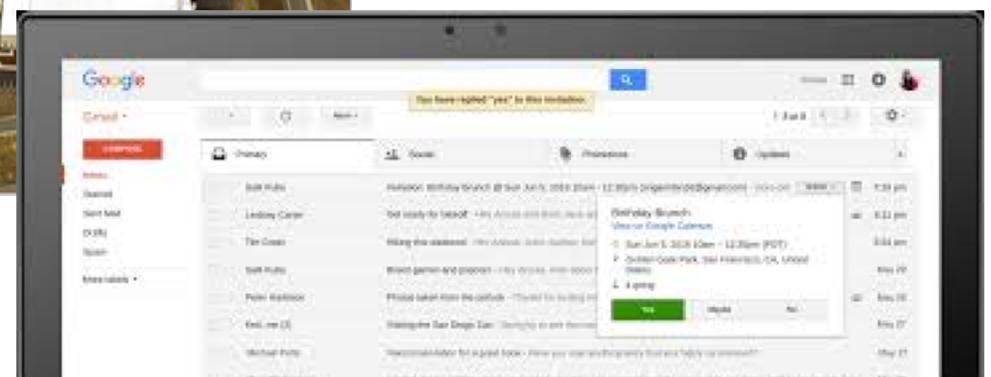
Google's Datacenter in Oregon



Πραγματικό Υλικό



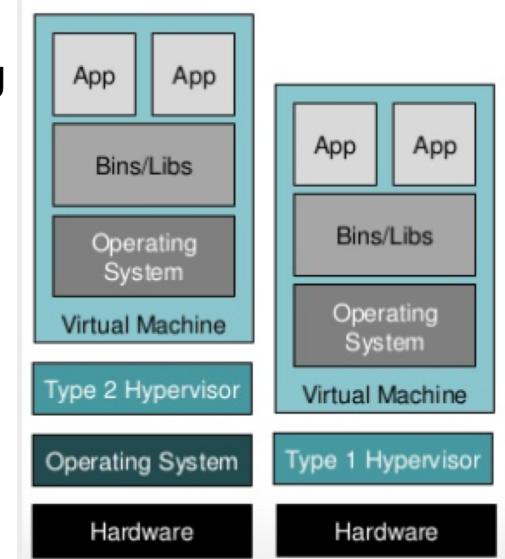
Εικονική Υπηρεσία
(Virtualized Service)



Hypervisors (Επόπτες) και Virtual Machines (VMs)



- **Hypervisors**: πρόγραμμα το οποίο δημιουργεί και εκτελεί **εικονικές μηχανές (virtual machines)**.
- **Type-1: native or bare-metal hypervisors**
 - Τρέχουν απευθείας πάνω στο υλικό. Κατάλληλα για διαθέτες (servers). Συνήθως προσαρμοσμένα LINUX kernels.
 - Παραδείγματα: Xen (open source), Oracle VM Server, Hyper-V (Microsoft), ESX/ESXi (VMWARE).
 - Γνωστά ως IaaS στην ορολογία του Cloud Computing
- **Type-2 or hosted hypervisors**
 - Τρέχουν ως πρόγραμμα (user-space) πάνω από το Kernel ενός άλλου Λ.Σ. (MacOSx, Windows, Linux)
 - Παραδείγματα: VMware Workstation, VMware Player, VirtualBox, Parallels Desktop for Mac and QEMU

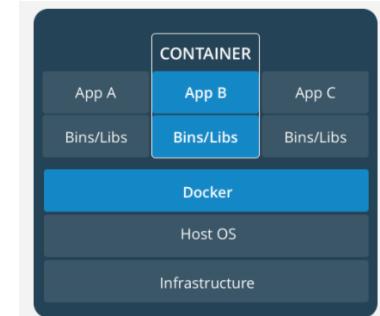
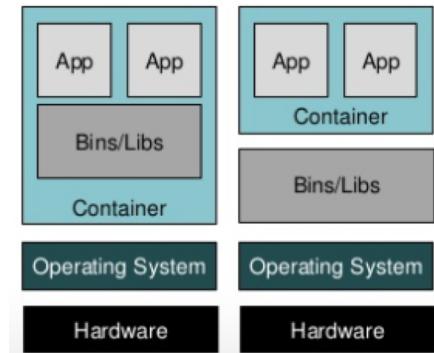




Virtual Private Server

- **Virtual Private Server (VPS)**: αξιοποιεί Εικονικοποίηση για να μοιράσει τον διαθέτη σε **containers**, το κάθε ένα εκ των οποίων λειτουργεί ως ξεχωριστός διαθέτης

- Κάθε VPS επιτρέπει root access, process management, stop/reboot VPS, κτλ.
- Παράδειγμα: **OpenVZ**, Solaris Containers & **LXC (Linux Containers)**
 - Στο μάθημα θα πάρετε 1 OpenVZ VPS με root δικαιώματα .
 - VPS δημιουργήθηκαν αρχικά για λόγους ασφάλειας (chroot jails)
 - Σήμερα είναι της μόδας software VPS τα οποία χρησιμοποιούνται για την διανομή λογισμικού (π.χ. Docker PaaS)
 - Ξεφεύγουμε από εικονικοποίηση Λ.Σ. και εισερχόμαστε στον χώρο της Εικονικοποίησης αψηλότερων επιπτέδων (εκτός ΕΠΛ421)





IaaS, PaaS, SaaS

EPL421

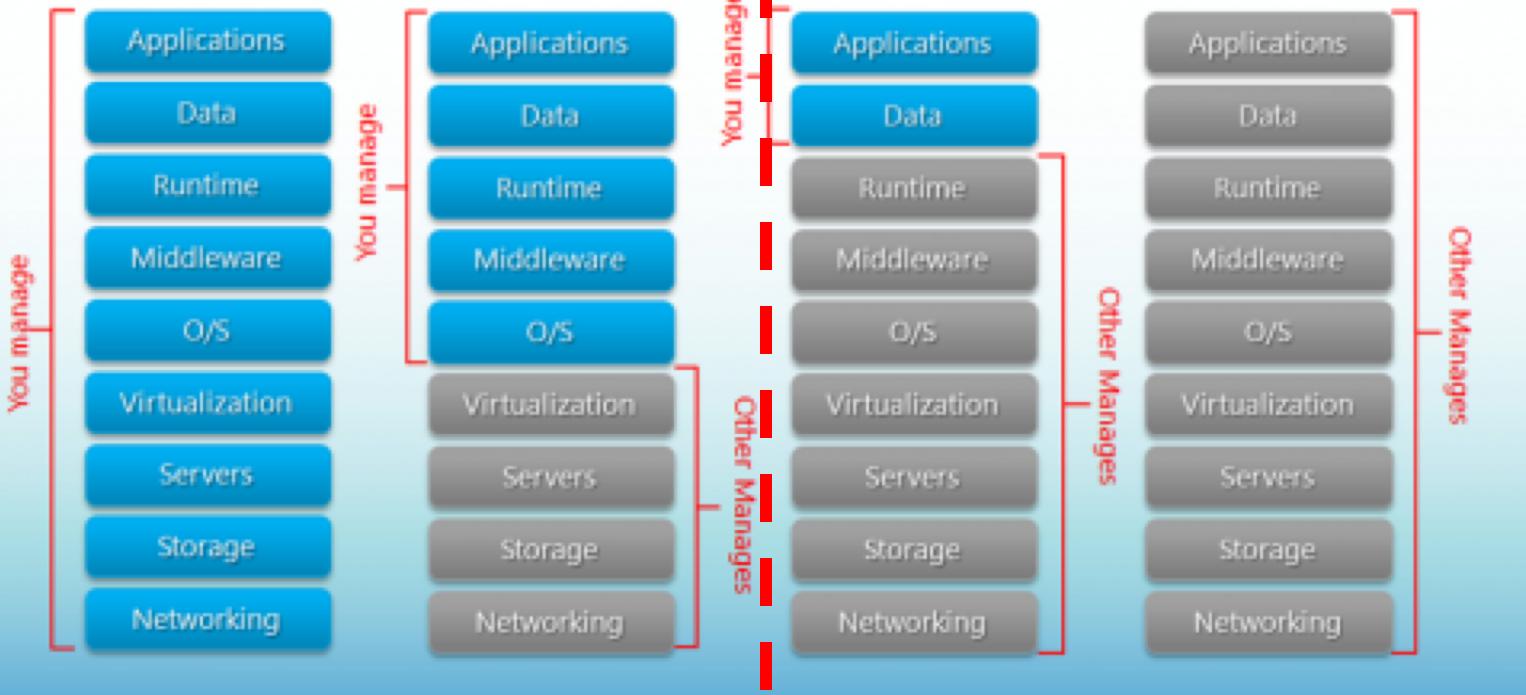
Separation of Responsibilities

On-Premises

e.g., Amazon
EC2
Infrastructure
(as a Service)

e.g.,
SQLserver
Platform
Azure
(as a Service)

e.g., GMAIL
for business
Software
(as a Service)





Finding Hardware Specs

```
<ada>$ cat /proc/cpuinfo
```

```
processor: 0
vendor_id: GenuineIntel
cpu family: 6
model: 13
model name: QEMU Virtual CPU version (cpu64-rhel6)
stepping: 3
microcode: 1
cpu MHz: 2400.084
cache size: 4096 KB
physical id: 0
siblings: 1
core id: 0
cpu cores: 1
apicid: 0
initial apicid: 0
fpu: yes
fpu_exception: yes
cpuid level: 4
wp: yes
```

```
# vmware -v
VMware ESX Server 3i 3.5.0 build-
158869
```



Περιεχόμενο Μαθήματος

- **Εβδ. 1-3: Εντολές UNIX και Εργαλεία Ωφελιμότητας (System Utilities)**
Εισαγωγή, Ιστορική Αναδρομή, Απλές και προχωρημένες εντολές, Κανονικές εκφράσεις, Ανακατεύθυνση (redirect) και διοχέτευση (pipe), Διαχείριση Αρχείων, Διεργασίες, Εντολές περιβάλλοντος, επικοινωνίας, εισόδου/εξόδου, ελέγχου διεργασιών, Εργαλεία Ωφελιμότητας: sed, awk κτλ.)
- **Εβδ. 4-5: Προγραμματισμός στο Κέλυφος (bash)**
Εισαγωγή, Μεταβλητές, Συνθήκες Ελέγχου (if, select), Επαναληπτικοί Βρόγχοι (for, while, until), Έλεγχος Ροής (break, continue), Συναρτήσεις, Πίνακες, λογικοί και αριθμητικοί τελεστές κ.α.



Περιεχόμενο Μαθήματος

- **Εβδ. 6: Συστήματα Αρχείων.**

Εισαγωγή στα Αρχεία και Συστήματα Αρχείων στο Unix (Τύποι Αρχείων, Partitions, i-nodes, blocks), Μέθοδοι Επεξεργασίας Αρχείων: Standard I/O vs. Χαμηλού Επιπέδου I/O, Μέτα-πληροφορίες Αρχείων, Δομή και Πρόσβαση, Διαχείριση Αρχείων με κλήσεις συστήματος, Διαχείριση Καταλόγων, Προσπέλαση Καταλόγων με συναρτήσεις βιβλιοθήκης

- **Εβδ. 7-8: Διεργασίες: Περιβάλλον, Έλεγχος και Σήματα**

Διεργασίες: Δομή & Αναπαράσταση, Ταυτότητες Διεργασιών, Δημιουργία Διεργασιών, Μεταβλητές Περιβάλλοντος, Ορφανές Διεργασίες, Διεργασίες και Αρχεία, Αναμονή Διεργασιών, Zombie Διεργασίες, Σήματα και Διεργασίες

MIDTERM



Περιεχόμενο Μαθήματος

- Εβδ. 9-10: Δια-διεργασιακή Επικοινωνία (IPC)**

Επικοινωνία μεταξύ Διεργασιών, Μελέτη τρόπων επικοινωνίας: i) Σωλήνες (Pipes), ii) FIFO (Named Pipes), iii) Ουρές Μηνυμάτων (Message Queues), iv) Κοινόχρηστη Μνήμη (Shared Memory) με Σηματοφόρους (Semaphores) και v) Προγραμματισμός Υποδοχών (Socket Programming).

- Εβδ. 11: Πολυνηματικός Προγραμματισμός (Multithreading)**

Πλεονεκτήματα Νημάτων, Είδη Νημάτων, Πολυνηματικά Μοντέλα, Η Βιβλιοθήκη <pthread.h> και παραδείγματα, Θέματα διαχείρισης ταυτοχρονίας, διαχείριση αδιεξόδων.

- Εβδ. 12: Παρουσιάσεις Φοιτητών**

Systems Programming in Windows: Sockets, Process Management, Threads and Concurrency, IPC

Scripting Languages: PHP, Ruby, Python, Perl, Smalltalk

- Εβδ. 13: Προχωρημένα Θέματα**

Map-Reduce, DevOps, κτλ.