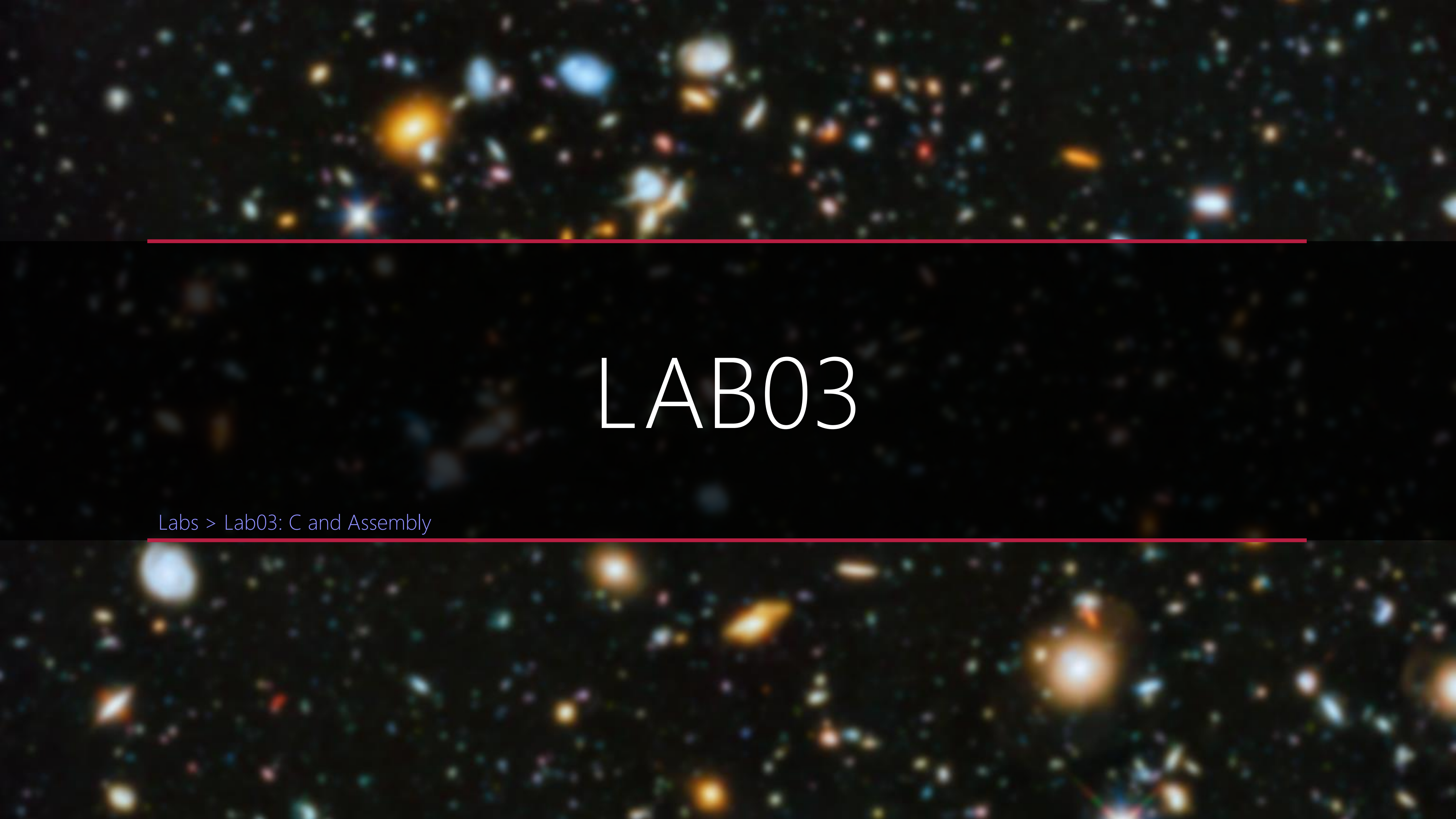
6

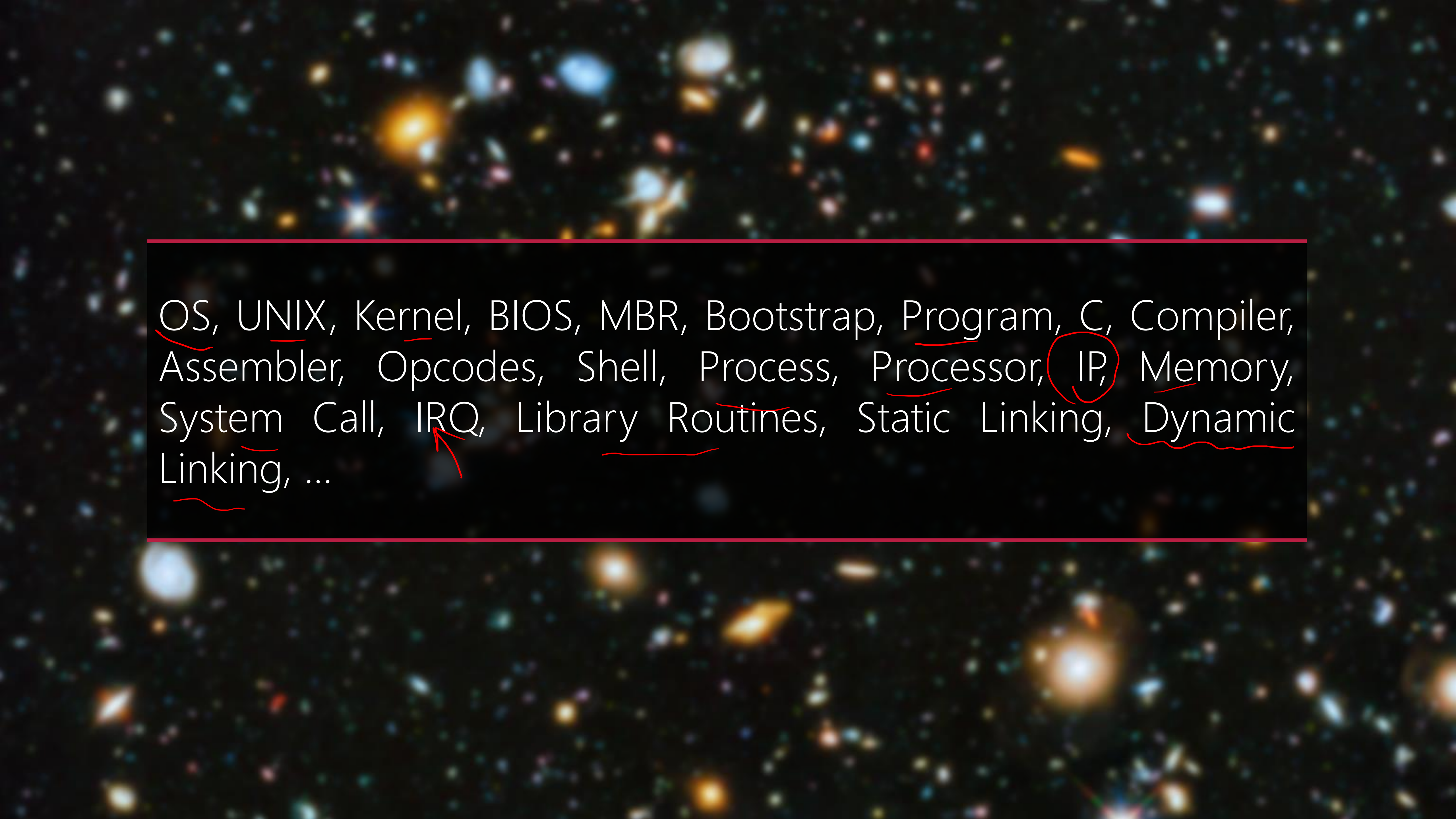


LAB03

Labs > Lab03: C and Assembly

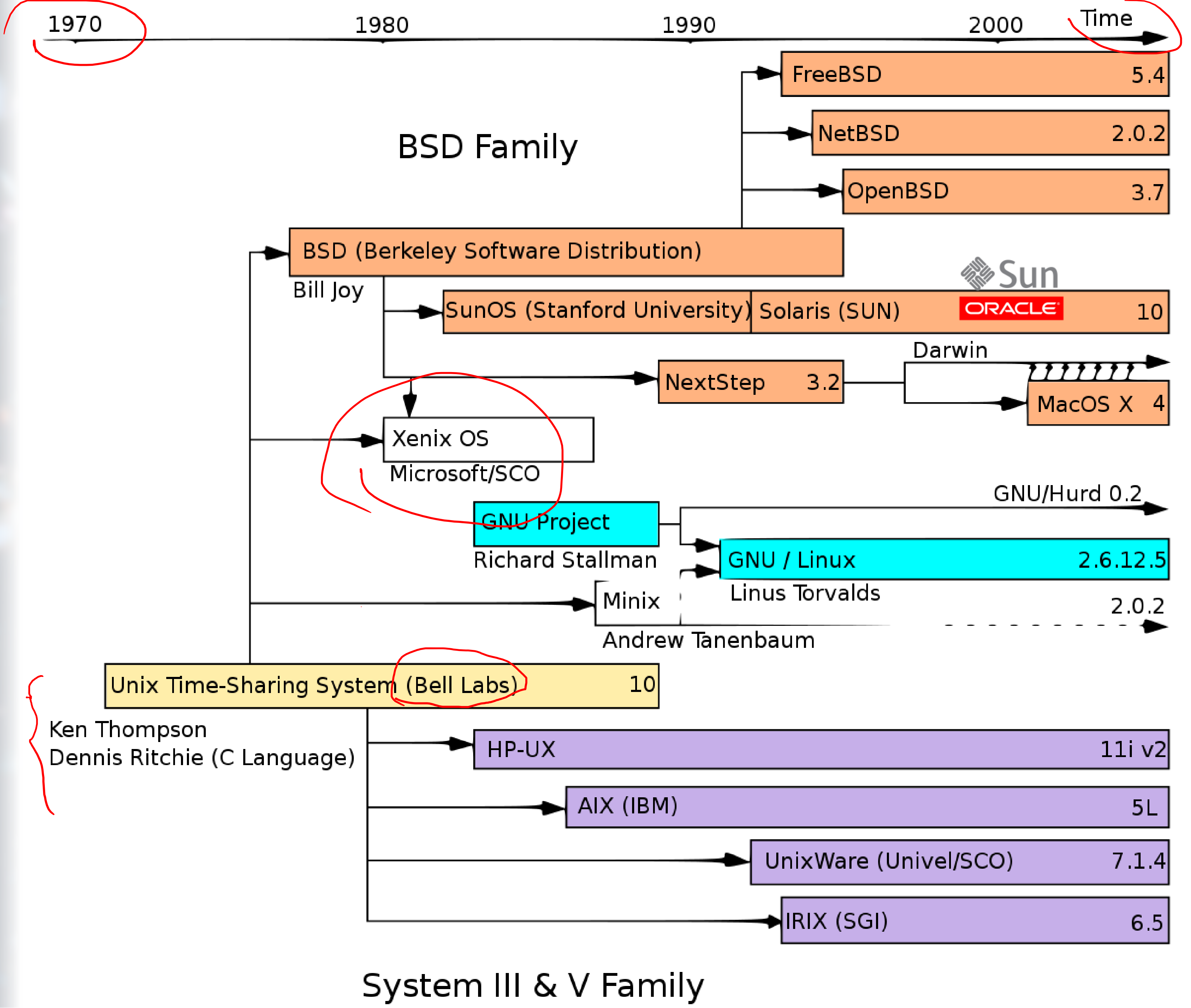
LEC03

Lecs > Lec03: Shell



OS, UNIX, Kernel, BIOS, MBR, Bootstrap, Program, C, Compiler, Assembler, Opcodes, Shell, Process, Processor, IP, Memory, System Call, IRQ, Library Routines, Static Linking, Dynamic Linking, …

HISTORY



1970 1980 1990 2000 Time

BSD Family

FreeBSD 5.4

NetBSD 2.0.2

OpenBSD 3.7

BSD (Berkeley Software Distribution)

Bill Joy

Sun

SunOS (Stanford University) Solaris (SUN) ORACLE 10

Darwin

NextStep 3.2

MacOS X 4

Xenix OS

Microsoft/SCO

GNU/Hurd 0.2

GNU Project

Richard Stallman

GNU / Linux 2.6.12.5

Linus Torvalds

Minix

2.0.2

Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

Ken Thompson
Dennis Ritchie (C Language)

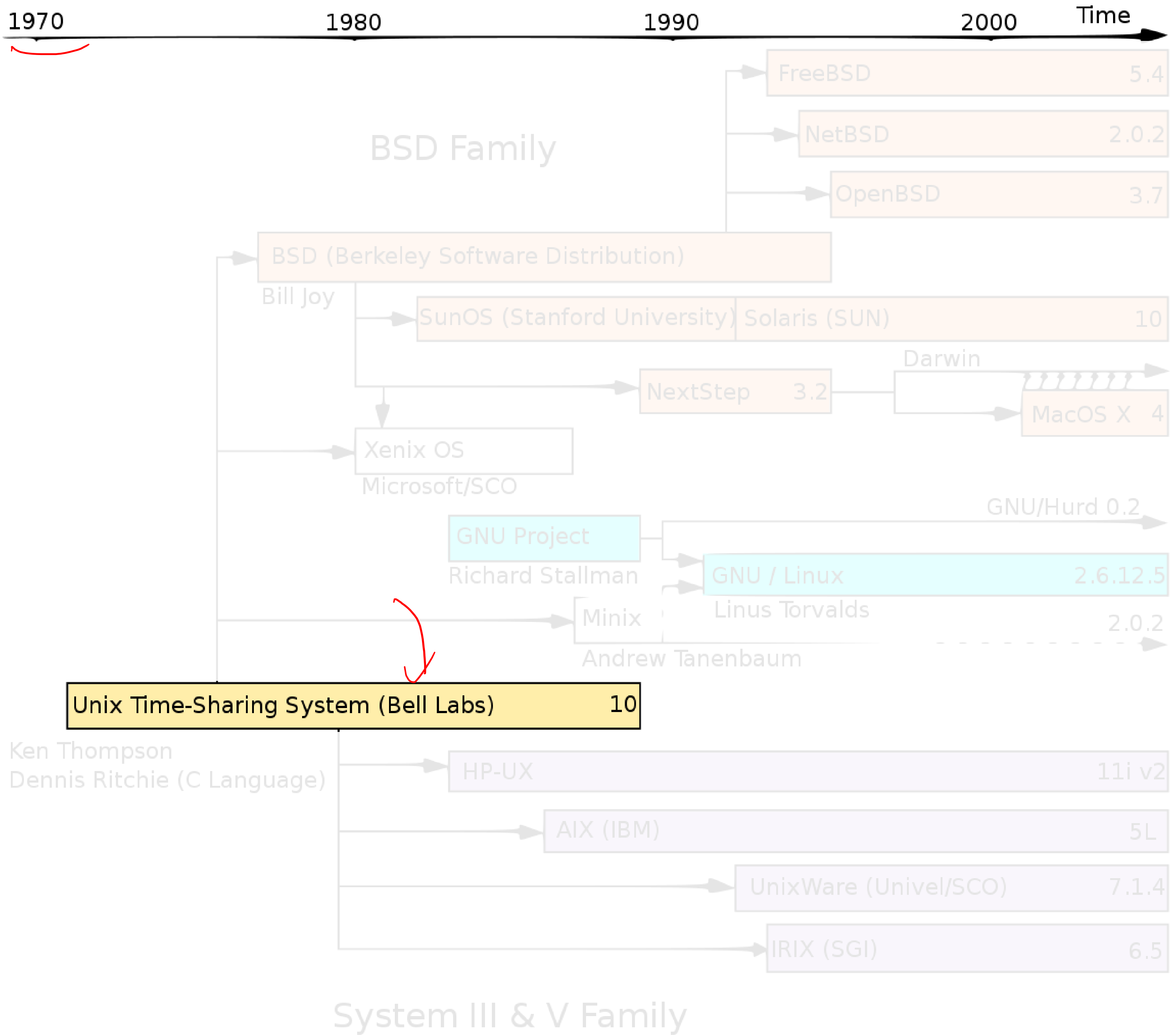
HP-UX 11i v2

AIX (IBM) 5L

UnixWare (Univel/SCO) 7.1.4

IRIX (SGI) 6.5

System III & V Family



1970
1980
1990
2000
Time

BSD Family

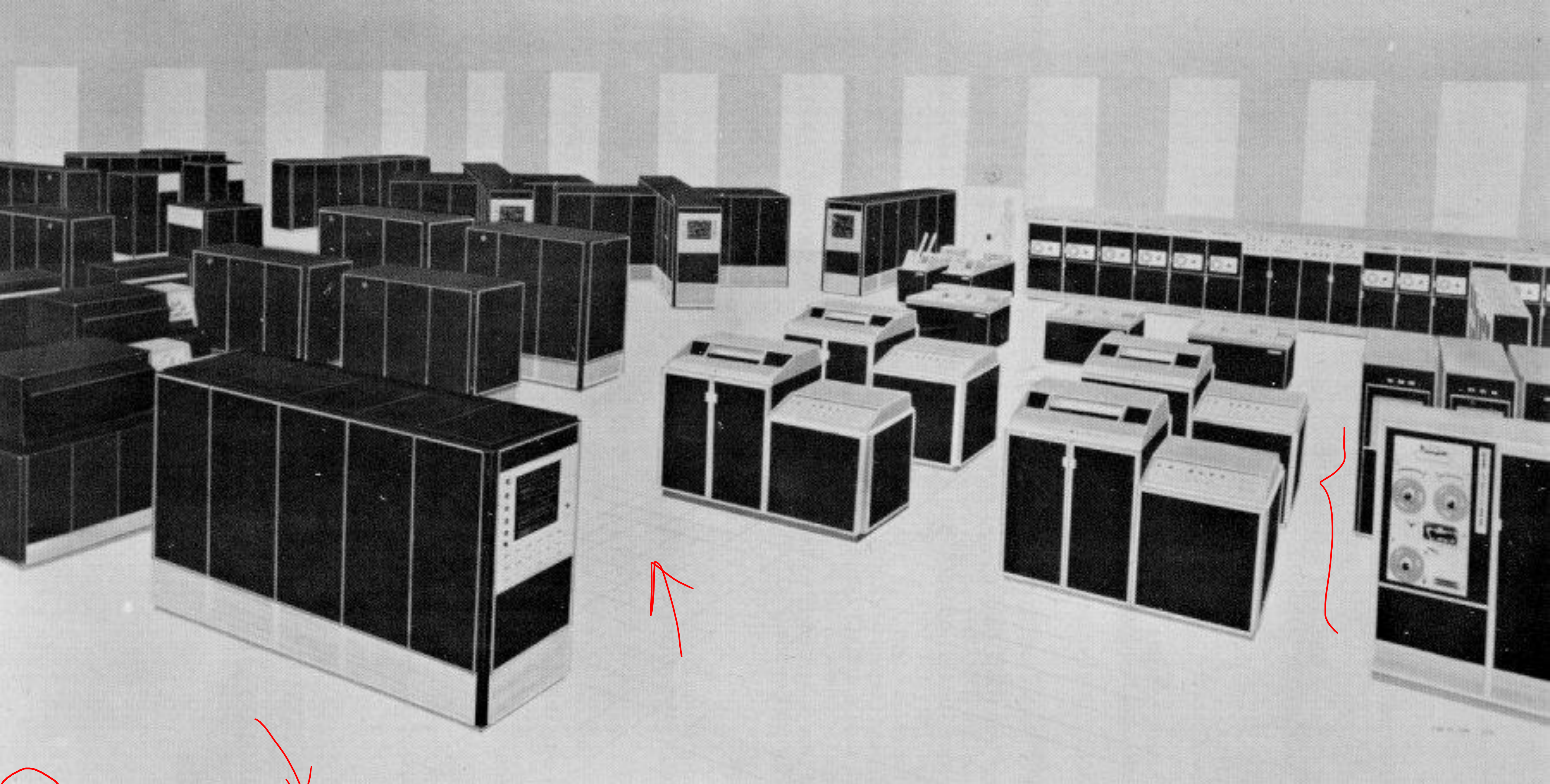
FreeBSD 5.4
NetBSD 2.0.2
OpenBSD 3.7
BSD (Berkeley Software Distribution)
Bill Joy
SunOS (Stanford University)
Solaris (SUN) 10
Darwin
NextStep 3.2
MacOS X 4
Xenix OS
Microsoft/SCO
GNU/Hurd 0.2
GNU Project
Richard Stallman
GNU / Linux 2.6.12.5
Linus Torvalds
Minix
2.0.2
Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

Ken Thompson
Dennis Ritchie (C Language)

HP-UX 11i v2
AIX (IBM) 5L
UnixWare (Univel/SCO) 7.1.4
IRIX (SGI) 6.5

System III & V Family



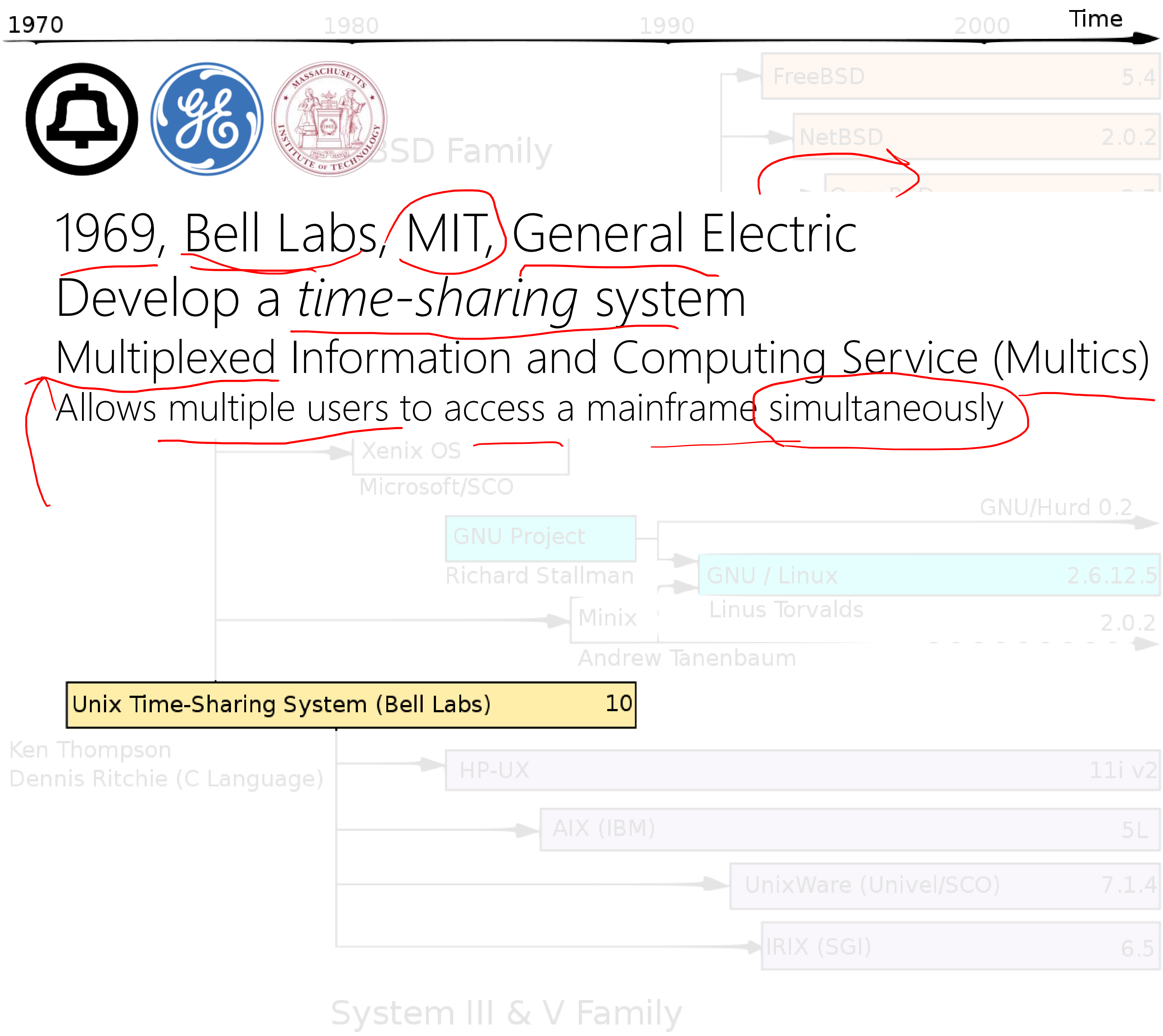
GE-645 mainframe

1969, Bell Labs, MIT, General Electric

Develop a *time-sharing* system

Multiplexed Information and Computing Service (Multics)

Allows multiple users to access a mainframe simultaneously



1970
1980
1990
2000
Time

FreeBSD 5.4
NetBSD 2.0.2
BSD Family

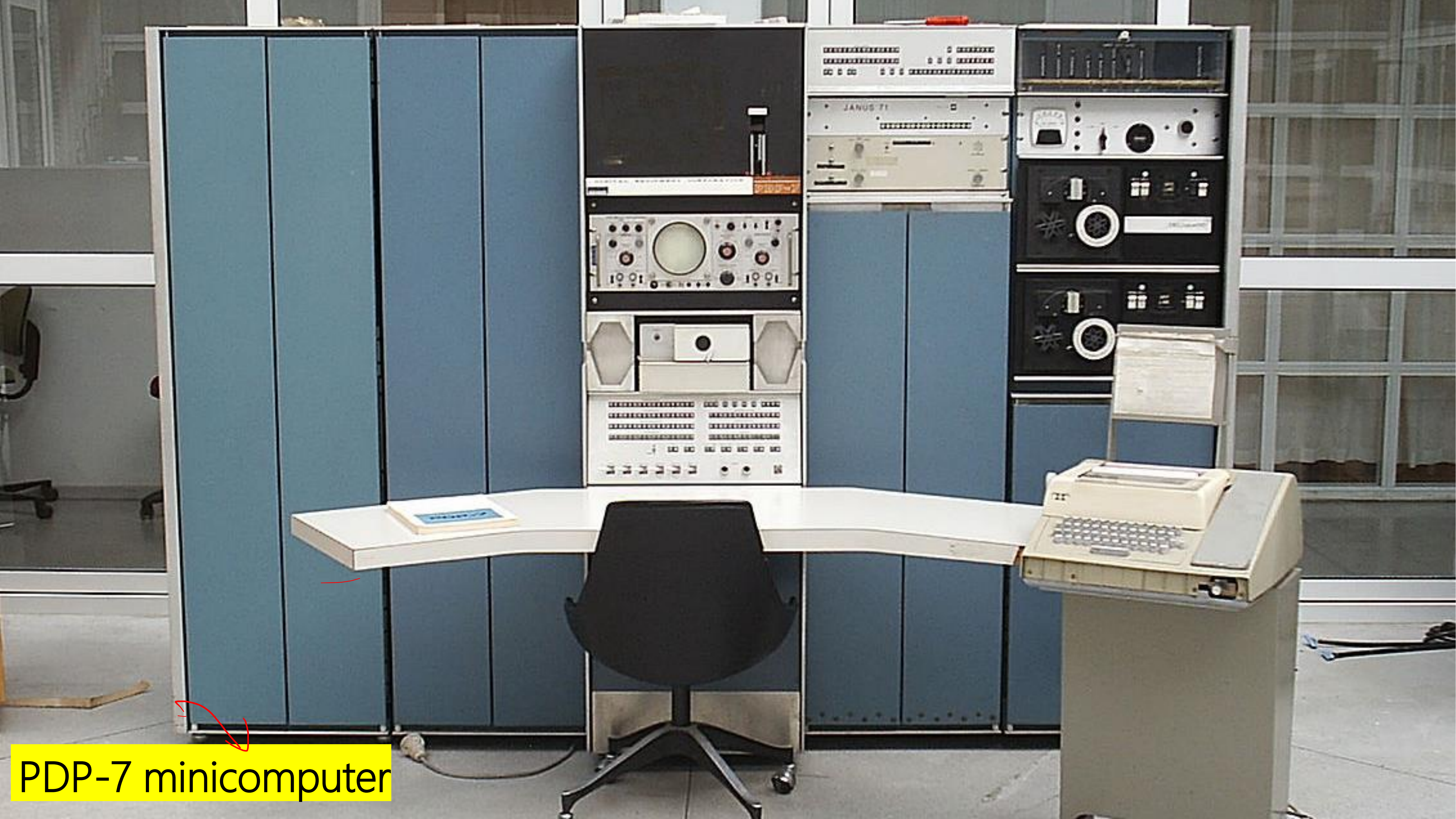
Xenix OS
Microsoft/SCO

GNU/Hurd 0.2
GNU Project
Richard Stallman
GNU / Linux 2.6.12.5
Linus Torvalds
Minix
Andrew Tanenbaum
2.0.2

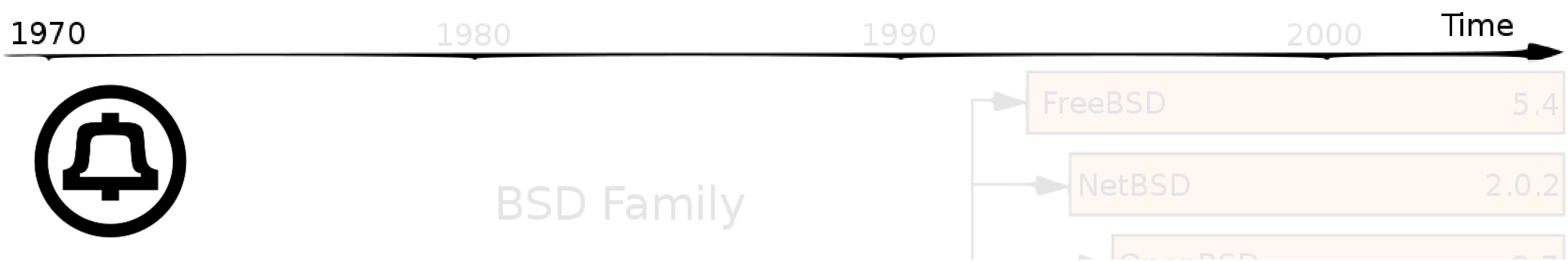
Unix Time-Sharing System (Bell Labs) 10
Ken Thompson
Dennis Ritchie (C Language)

HP-UX 11i v2
AIX (IBM) 5L
UnixWare (Univel/SCO) 7.1.4
IRIX (SGI) 6.5

System III & V Family



PDP-7 minicomputer



1969, Bell Labs

Ken Thompson & Denise Ritchie

Rewrite Multics but simpler

Uniplexed Information and Computing Service (Unics) , as "eunuchs"

UNIX

+ an assembler, editor, shell

Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

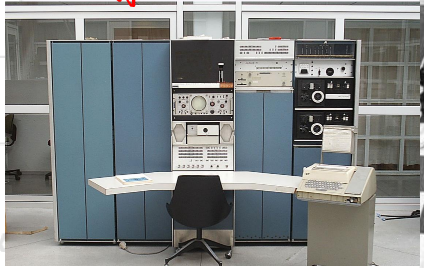
Ken Thompson

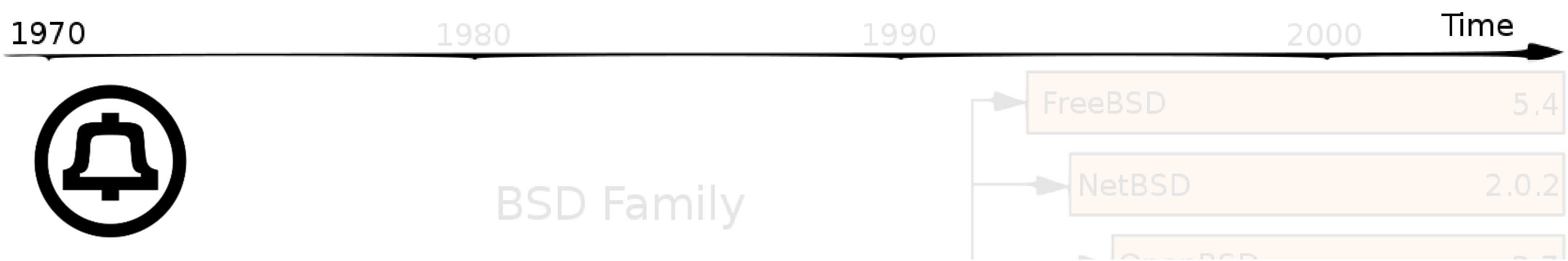
Den

PDP-7 minicomputer

AIX (IBM)

System III &





1973, Bell Labs

Ken Thompson & Denise Ritchie

Rewrite UNIX but not in assembly

High-Level Programming Language

C Programming Language

UNIX v.4.0

C C+ASS_+

PDP-7 minicomputer

1970 1980 1990 2000 Time

FreeBSD 5.4

NetBSD 2.0.2

BSD Family

1978, Bell Labs

Bell was not able to make UNIX as product. Why?

Thompson quietly began shipping out tapes and disks, signed, *"Love, Ken"*

GNU Project

Richard Stallman

GNU / Linu

Linus Torva

Minix

Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

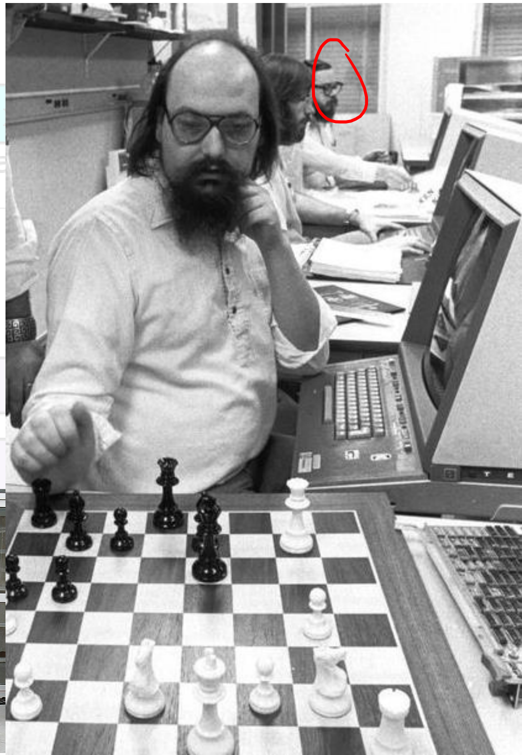
Ken Thompson

Den

PDP-7 minicomputer

AIX (IBM)

System III &

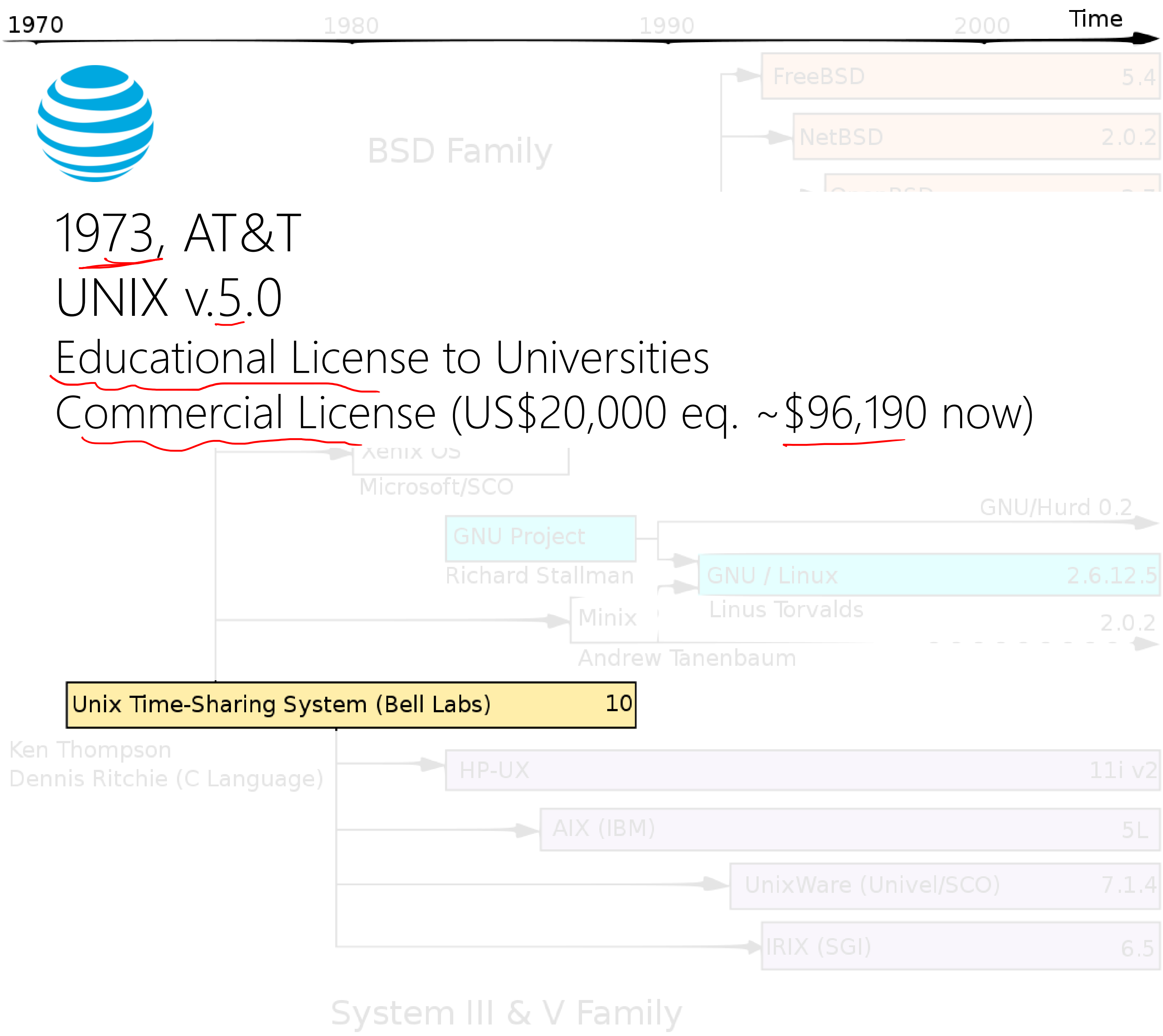


1973, AT&T

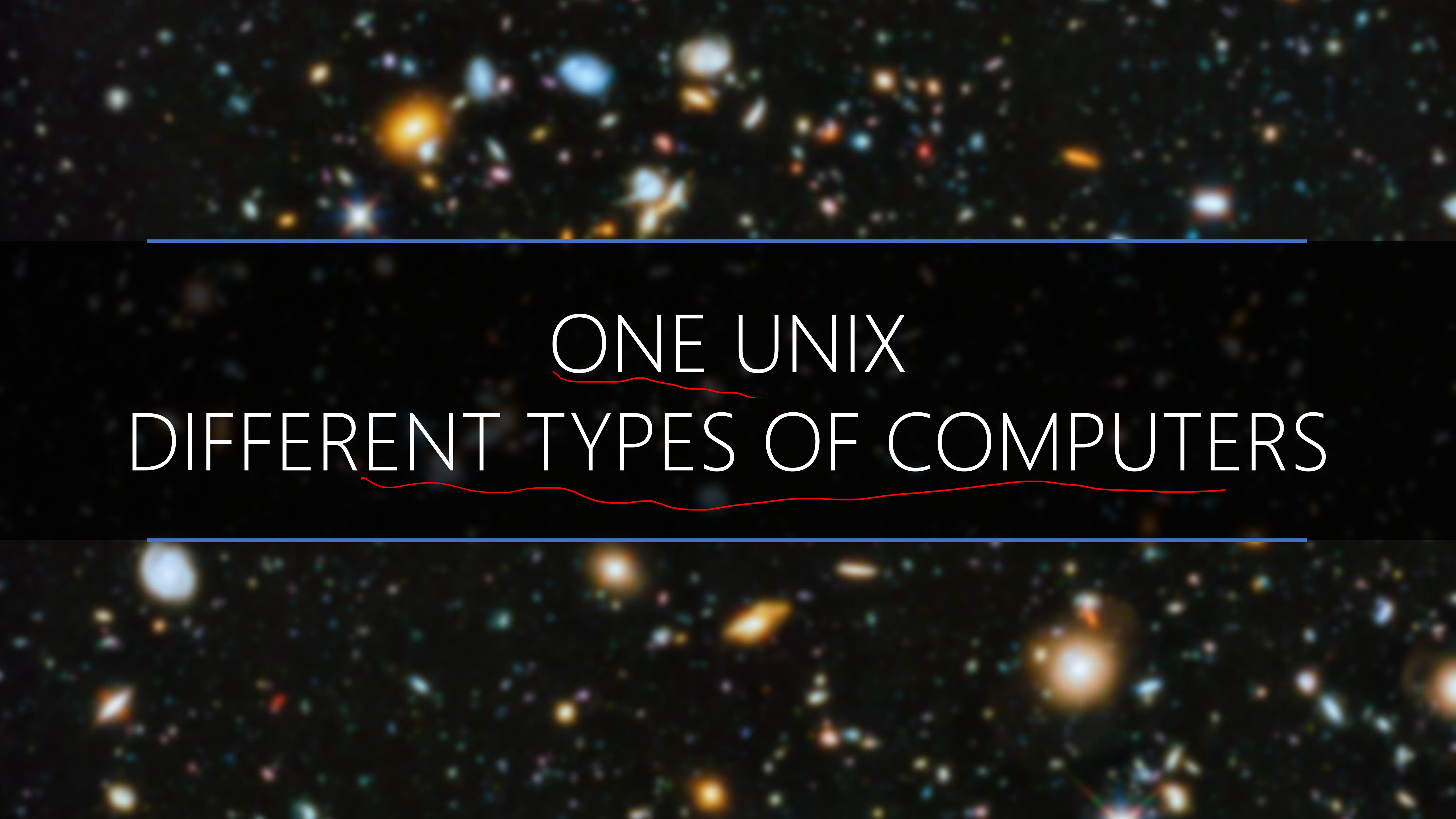
UNIX v.5.0

Educational License to Universities

Commercial License (US$20,000 eq. ~$96,190 now)

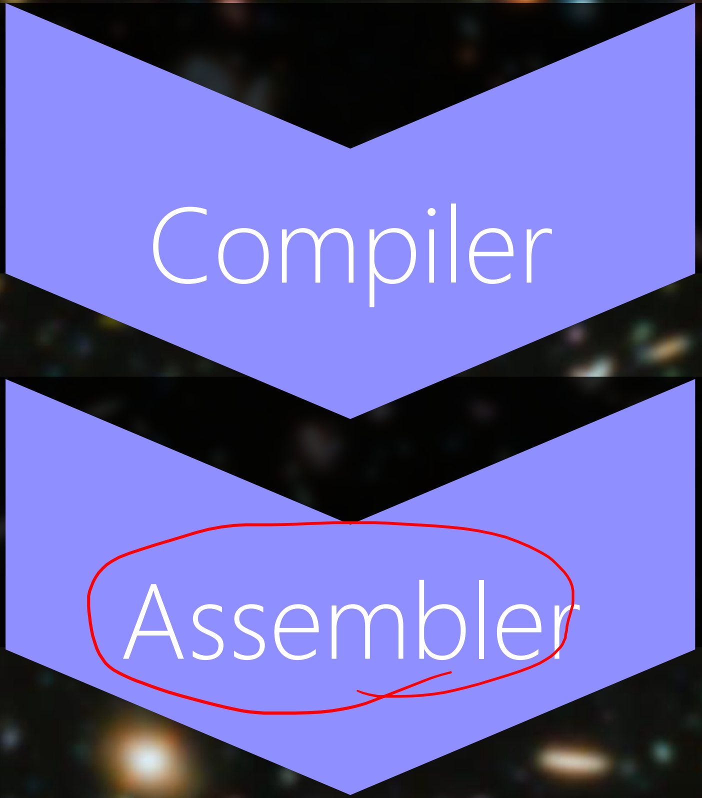


USENIX: UNIX Users, New York, 1974



ONE UNIX
DIFFERENT TYPES OF COMPUTERS

C



Compiler

Assembler

UNIX Source Code

Assembly

```
<printf@plt>:
jmpq            *0x3002(%rip)
pushq           $0x0
jmpq            401000 <.plt>
<main>:
push            %rbp
mov             %rsp,%rbp
lea             0xfd5(%rip),%rdi
mov             $0x0,%eax
callq           401010 <printf@plt>
nop
pop             %rbp
retq
```

OP Code

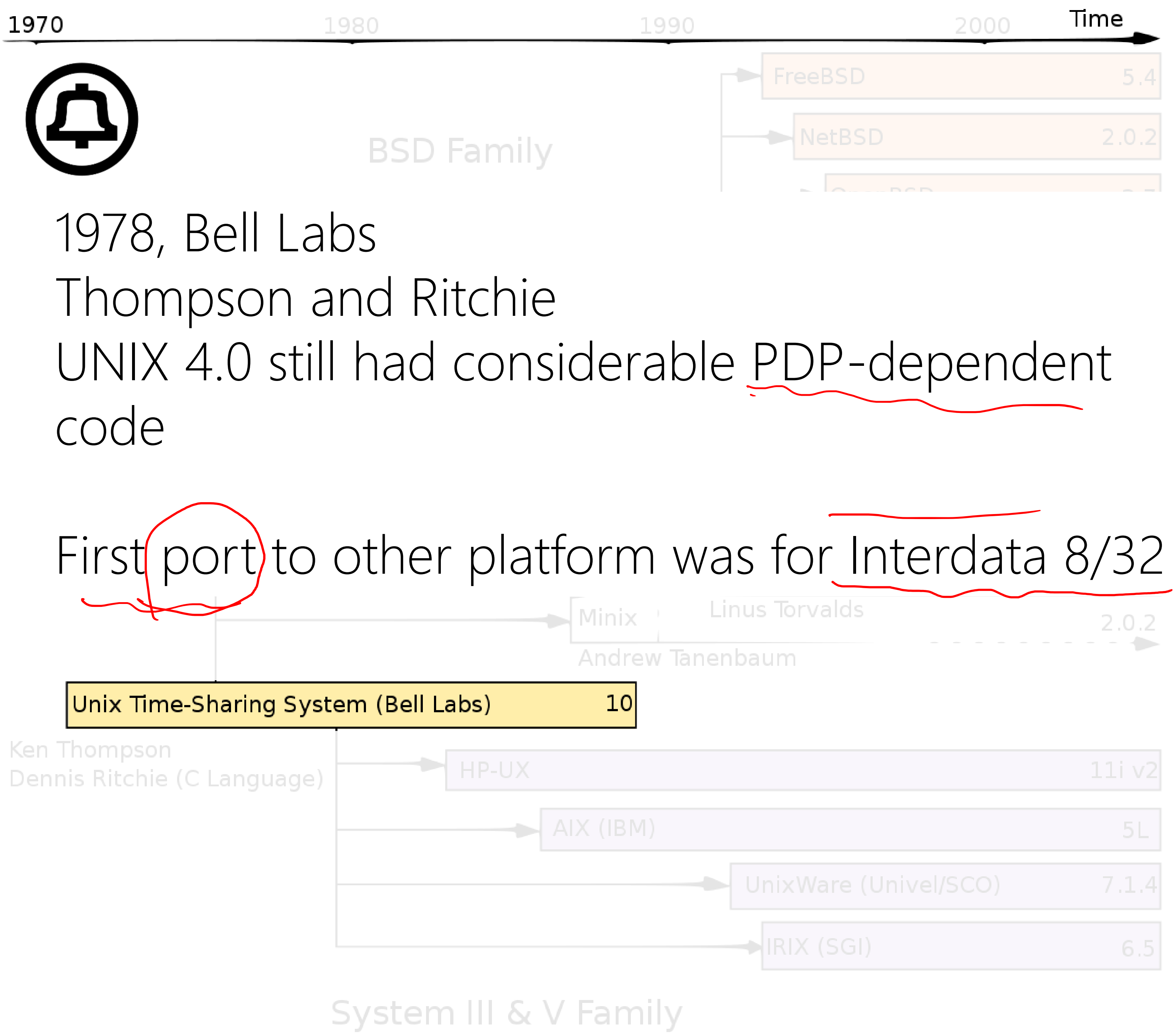
```
0001 0000 0000 0000 0000 0000 0000 0000
0011 0000 0003 0000 0000 0000 0000 0000
0000 0000 0000 0000 3309 0000 0000 0000
0096 0000 0000 0000 0000 0000 0000 0000
0001 0000 0000 0000 0000 0000 0000 0000
```

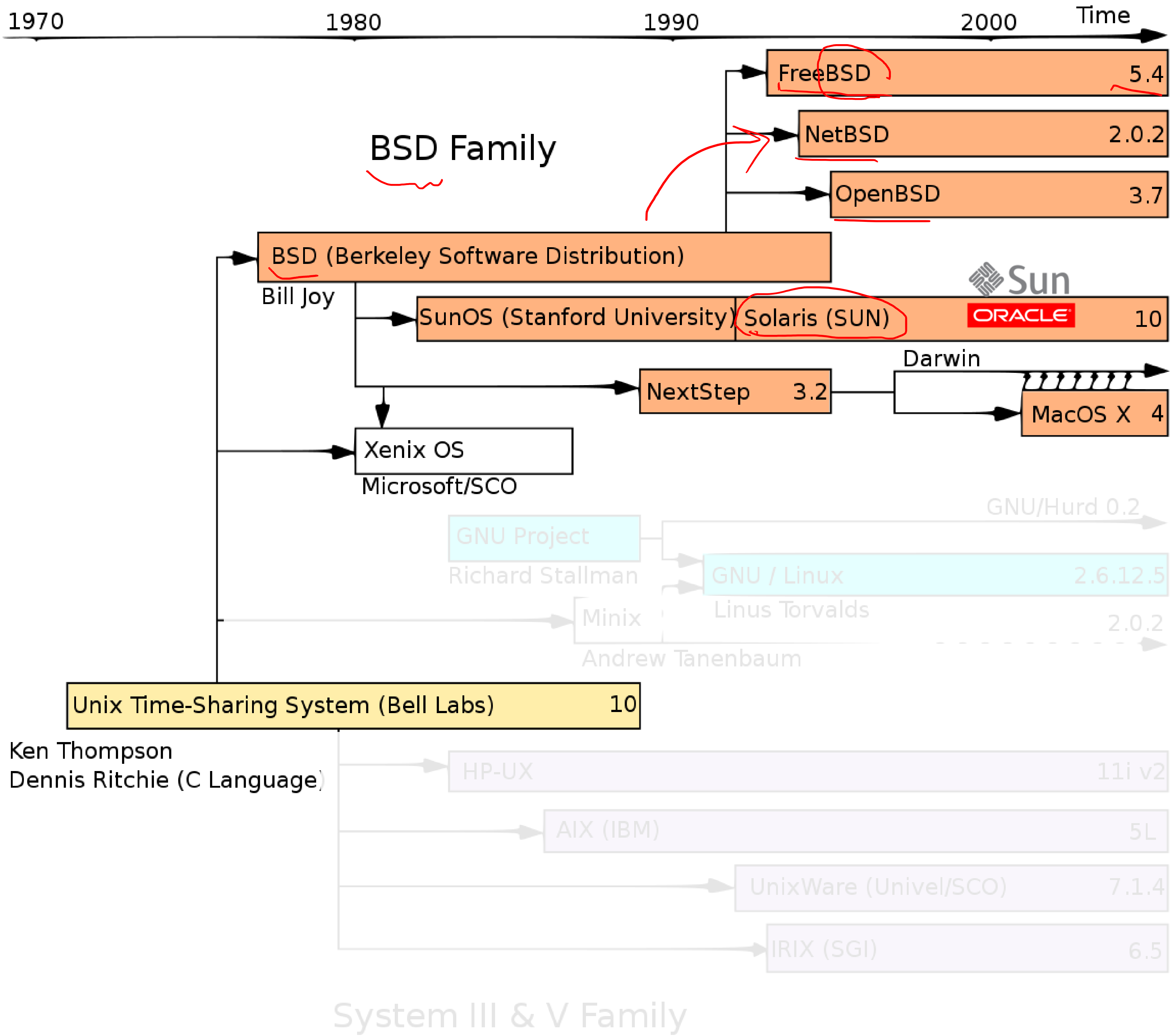
1978, Bell Labs

Thompson and Ritchie

UNIX 4.0 still had considerable PDP-dependent code

First port to other platform was for Interdata 8/32





1970 1980 1990 2000 Time

BSD Family

FreeBSD 5.4

NetBSD 2.0.2

OpenBSD 3.7

BSD (Berkeley Software Distribution)

Bill Joy

SunOS (Stanford University) Solaris (SUN) Sun ORACLE 10

Darwin

NextStep 3.2

MacOS X 4

Xenix OS

Microsoft/SCO

GNU/Hurd 0.2

GNU Project

Richard Stallman

GNU / Linux 2.6.12.5

Linus Torvalds

Minix 2.0.2

Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

Ken Thompson

Dennis Ritchie (C Language)

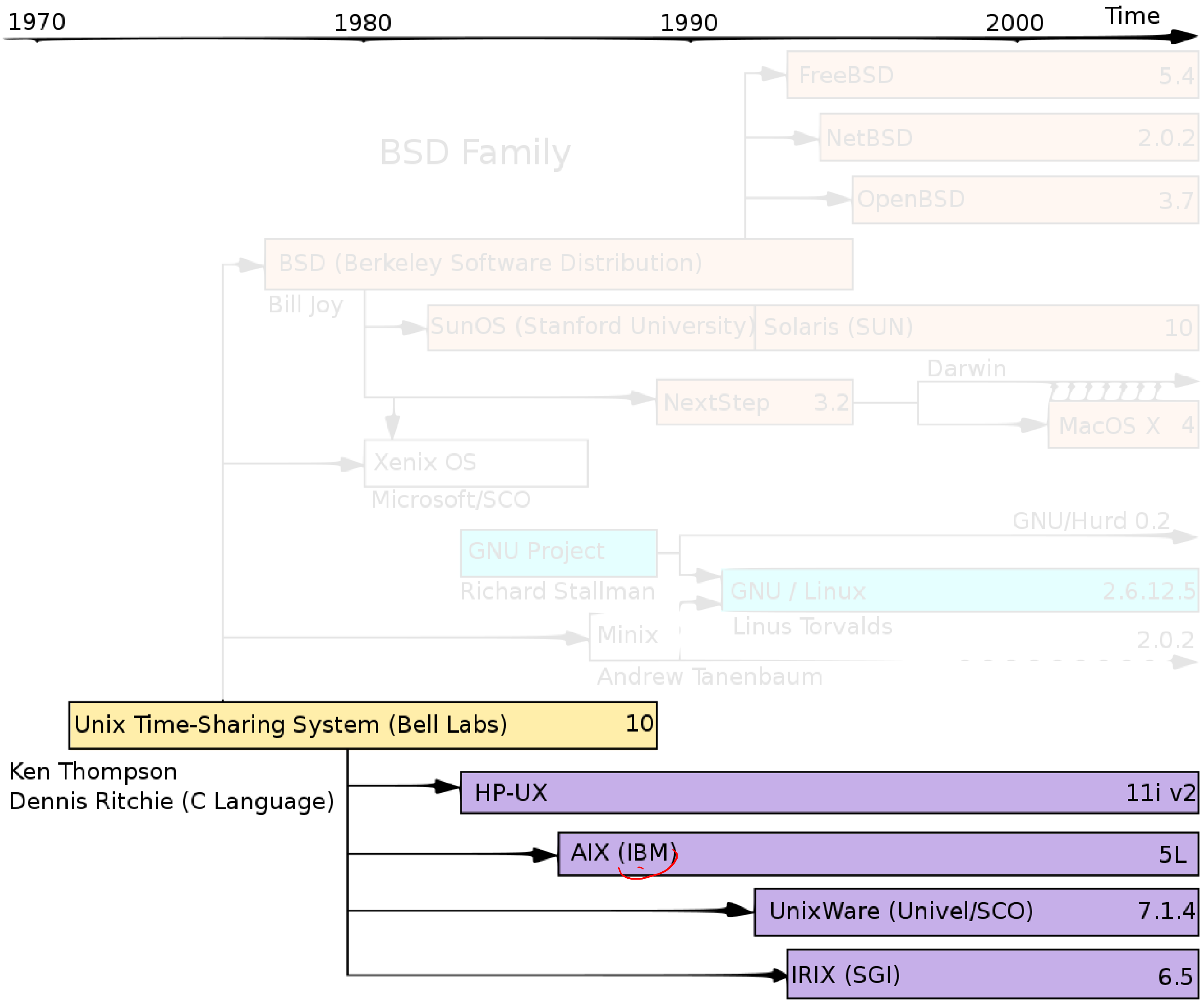
HP-UX 11i v2

AIX (IBM) 5L

UnixWare (Univel/SCO) 7.1.4

IRIX (SGI) 6.5

System III & V Family



1970 1980 1990 2000 Time

BSD Family

FreeBSD 5.4

NetBSD 2.0.2

OpenBSD 3.7

BSD (Berkeley Software Distribution)

Bill Joy

SunOS (Stanford University) Solaris (SUN) 10

Darwin

NextStep 3.2

MacOS X 4

Xenix OS

Microsoft/SCO

GNU/Hurd 0.2

GNU Project

Richard Stallman

GNU / Linux 2.6.12.5

Linus Torvalds

Minix 2.0.2

Andrew Tanenbaum

Unix Time-Sharing System (Bell Labs) 10

Ken Thompson

Dennis Ritchie (C Language)

HP-UX 11i v2

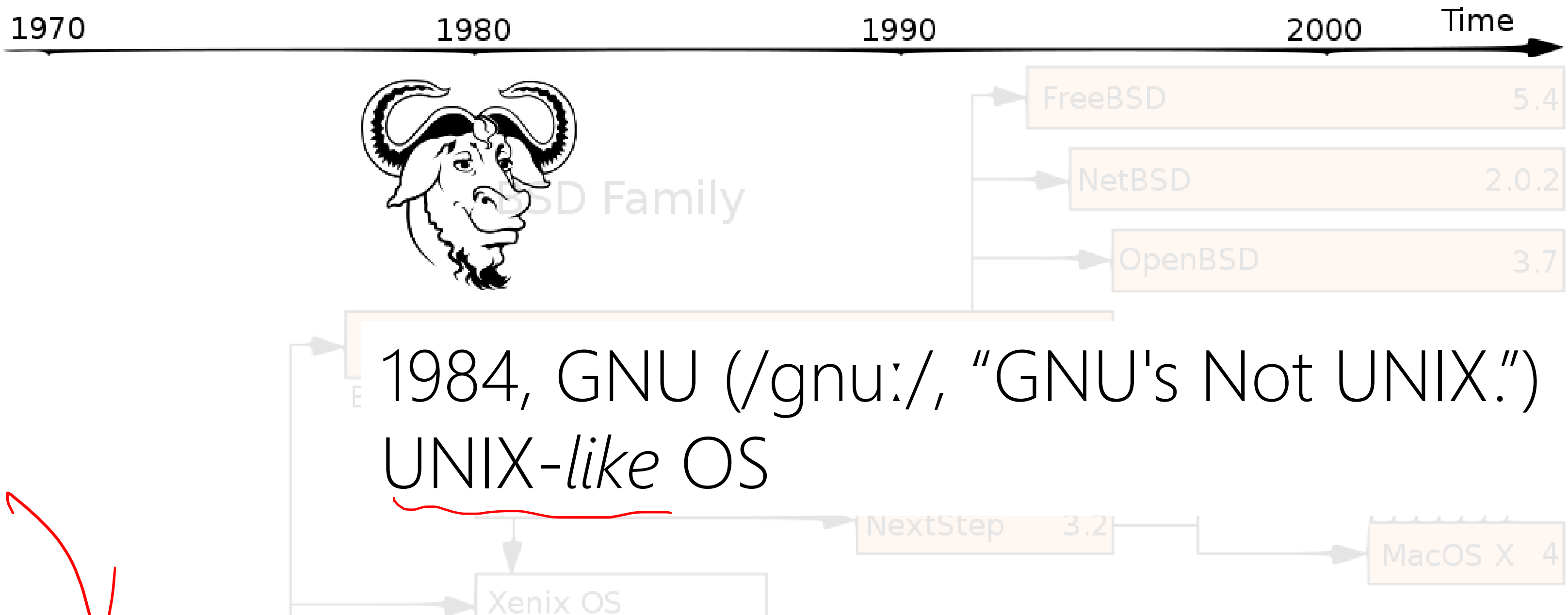
AIX (IBM) 5L

UnixWare (Univel/SCO) 7.1.4

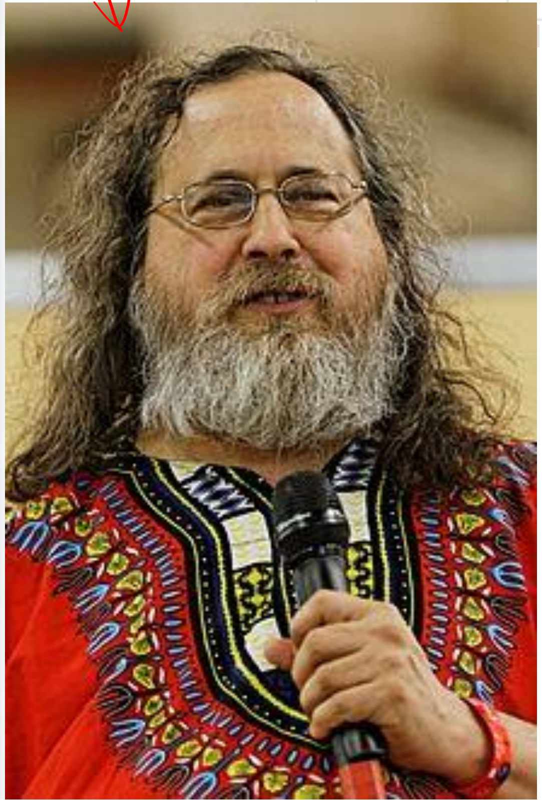
IRIX (SGI) 6.5

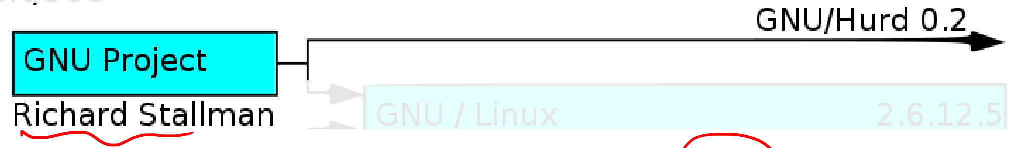
System III & V Family

PLATFORM-FREE UNIX



1984, GNU (/gnu:/, “GNU's Not UNIX.”) UNIX-*like* OS

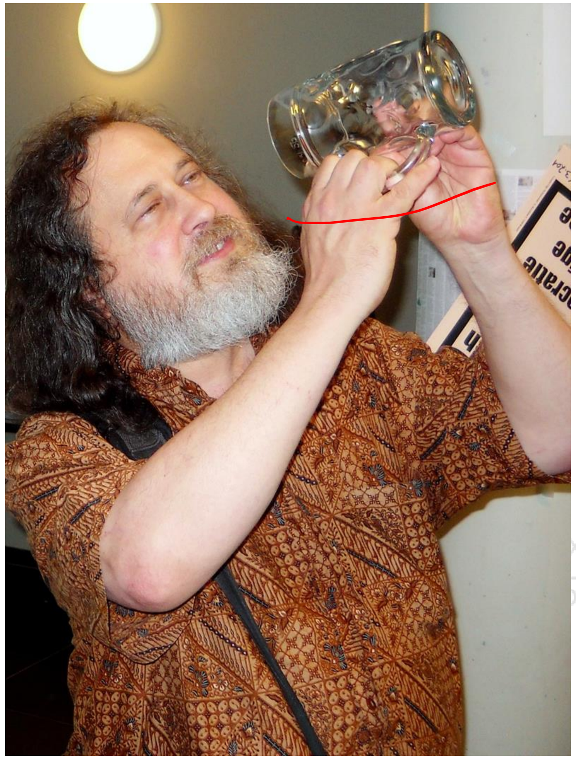




Recursive acronym for “GNU's Not UNIX.” “GNU” is pronounced g'noo, as one syllable, like saying “grew” but replacing the r with n.

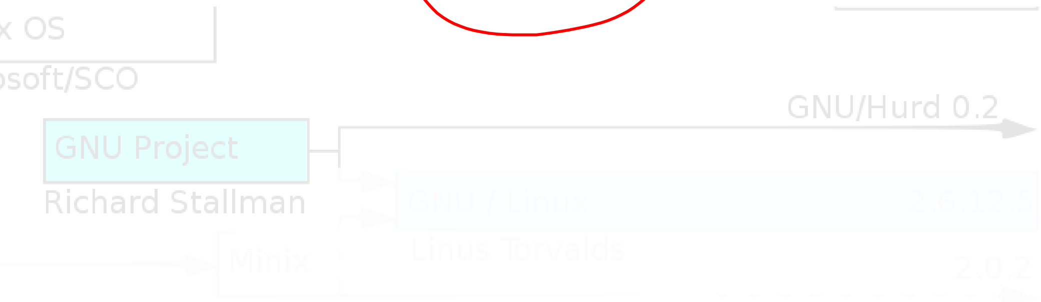
Code licensed under the GNU GPL can be reused in other computer programs as long as they also are released under the same or a compatible license.

Open Source Community



gratis vs. libre

Free Software vs. Libre Software



"Think free as in free speech, not free beer"

"Free as in freedom, not free as in free beer"

Access to the source code?

Make modifications?

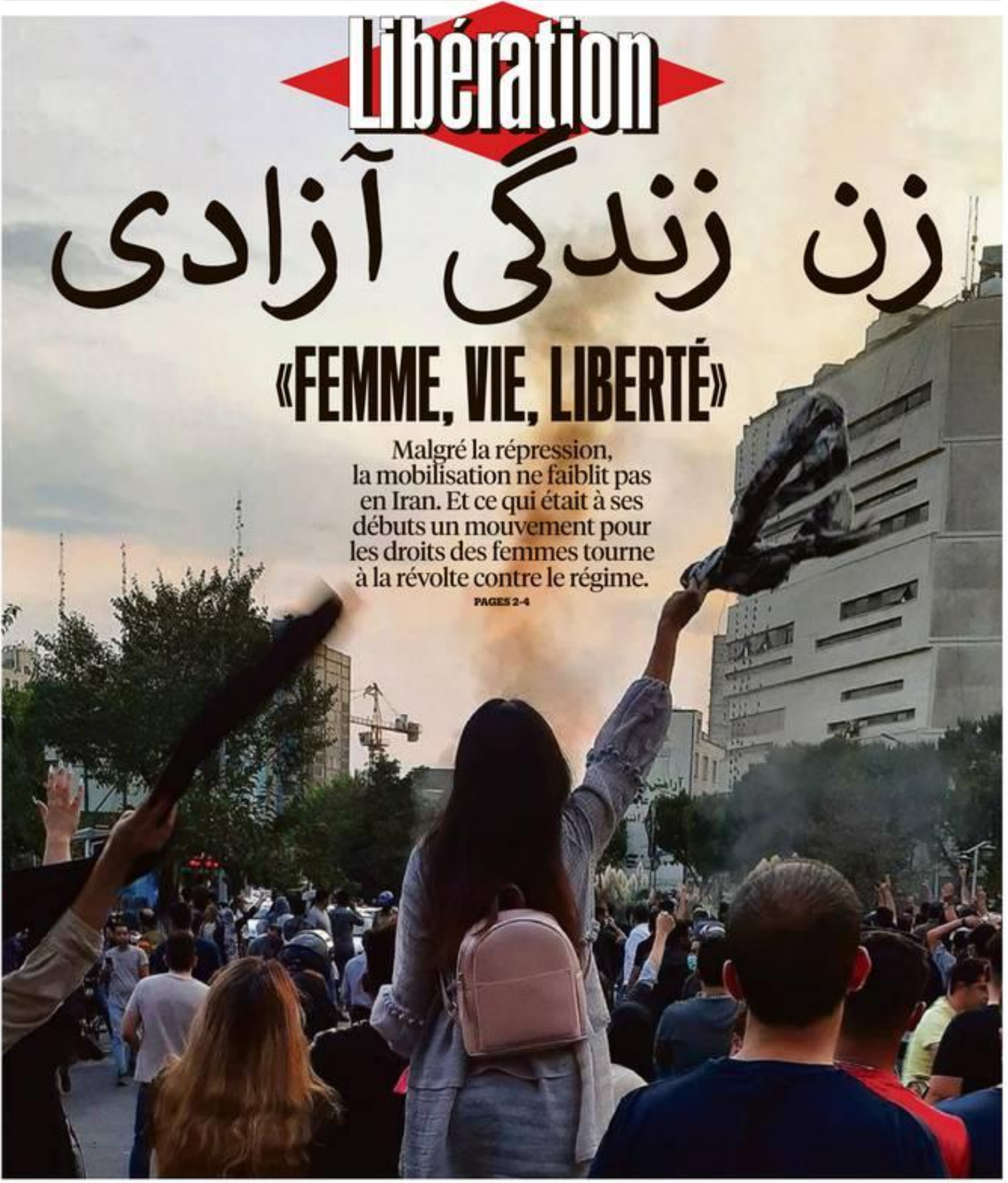
Make money?

2,50 € *Première édition.* Nº 12834 LUNDI 26 SEPTEMBRE 2022 www.liberation.fr

CÉCILE DUFLOT, JEAN-LUC MÉLENCHON, LAURENT BERGER…
Leurs propositions pour un budget plus vert et solidaire

PAGES 6-9

Libération



زن زندگی آزادی

«FEMME, VIE, LIBERTÉ»

Malgré la répression, la mobilisation ne faiblit pas en Iran. Et ce qui était à ses débuts un mouvement pour les droits des femmes tourne à la révolte contre le régime.

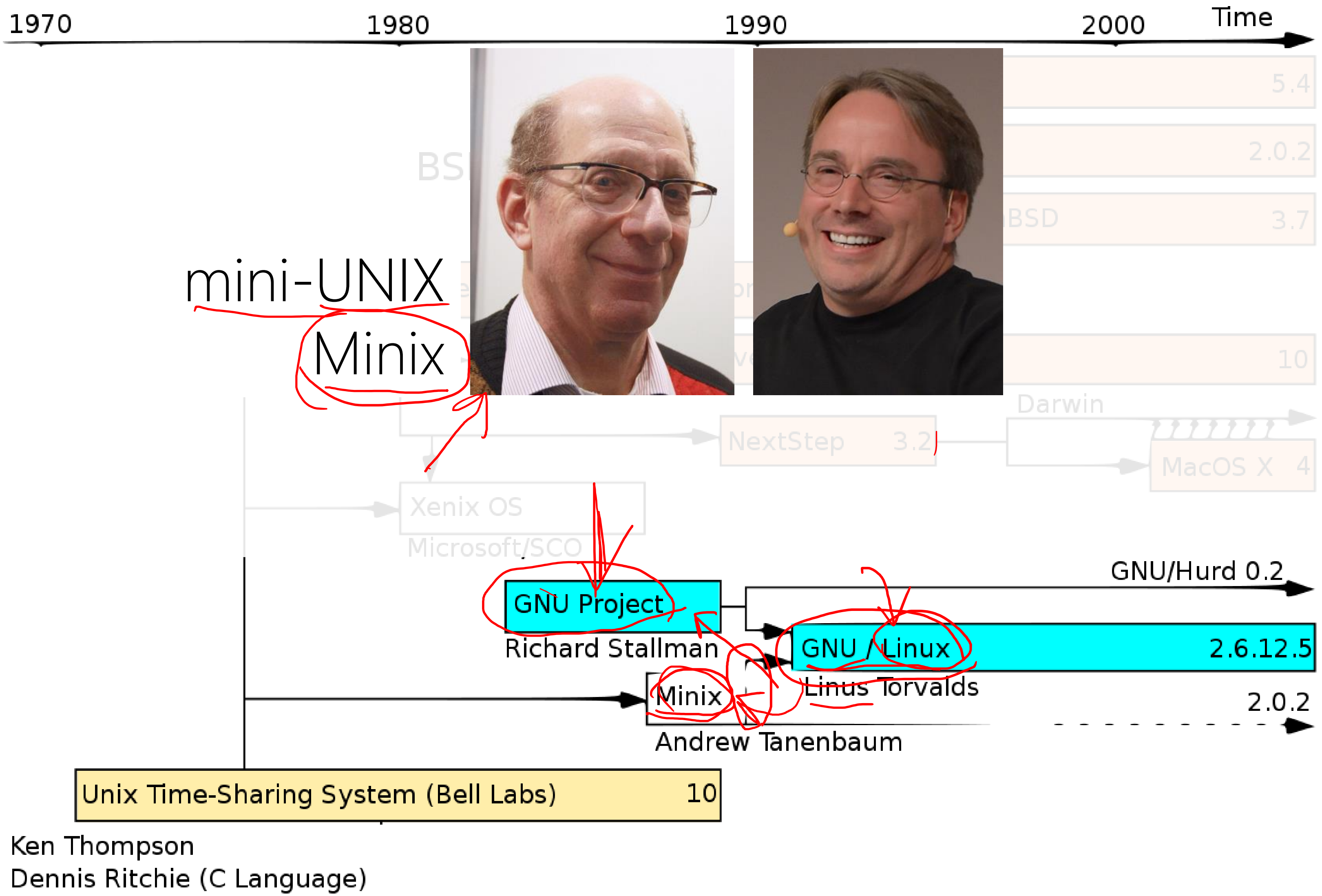
PAGES 2-4

A Téhéran, le 20 septembre. PHOTO SALAMPIX. ABACA

Libération

M 00135 - 926 - F: 2,50 €

IMPRIMÉ EN FRANCE / PRINTED IN FRANCE Algérie 1,60 €. Allemagne 3,00 €. Andorre 3,00 €. Belgique 2,50 €. Canada 5,00 $. DOM 3,00 €. Espagne 3,00 €. Etats-Unis 5,00 $. Grande-Bretagne 2,80 £. Grèce 3,00 €. Italie 3,00 €. Liban 7500 LBP. Luxembourg 2,50 €. Maroc 27 Dh. Pays-Bas 3,00 €. Portugal (continental) 3,40 €. Suisse 3,40 FS. Suisse alémanique 3,40 FS. Tunisie 8,00 DT. Zone CFA 2500 CFA



1970 1980 1990 2000 Time

mini-UNIX

Minix

BS

5.4

2.0.2

BSD 3.7

10

Darwin

NextStep 3.2

MacOS X 4

Xenix OS

Microsoft/SCO

GNU/Hurd 0.2

GNU Project

Richard Stallman

GNU / Linux 2.6.12.5

Linus Torvalds

Minix

Andrew Tanenbaum

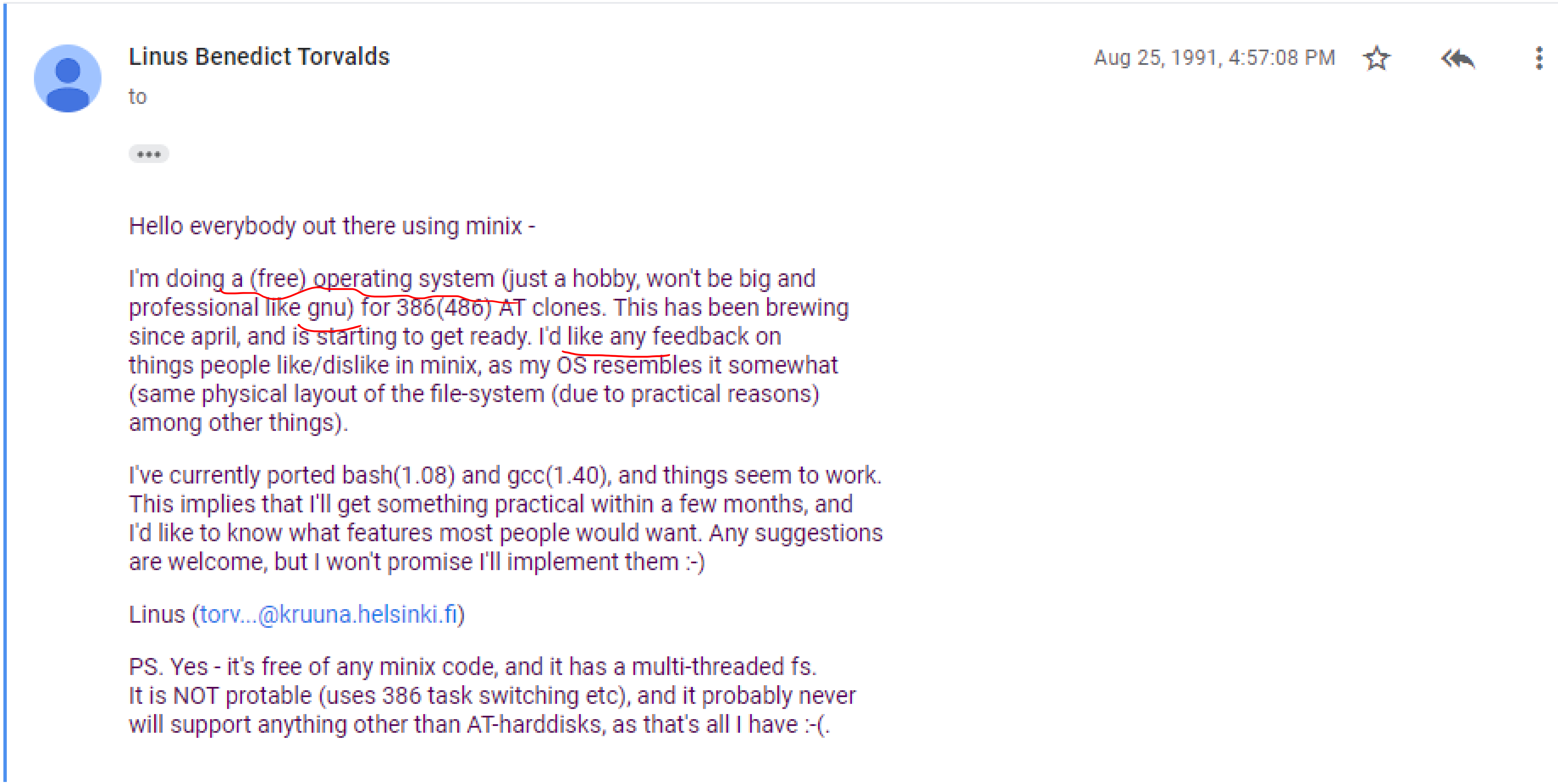
2.0.2

Unix Time-Sharing System (Bell Labs) 10

Ken Thompson

Dennis Ritchie (C Language)

In 1991, while attending the University of Helsinki Frustrated by the licensing of MINIX, limited it to educational use only, he began to work on his own operating system, which became the Linux.



Linus Benedict Torvalds

to

Aug 25, 1991, 4:57:08 PM

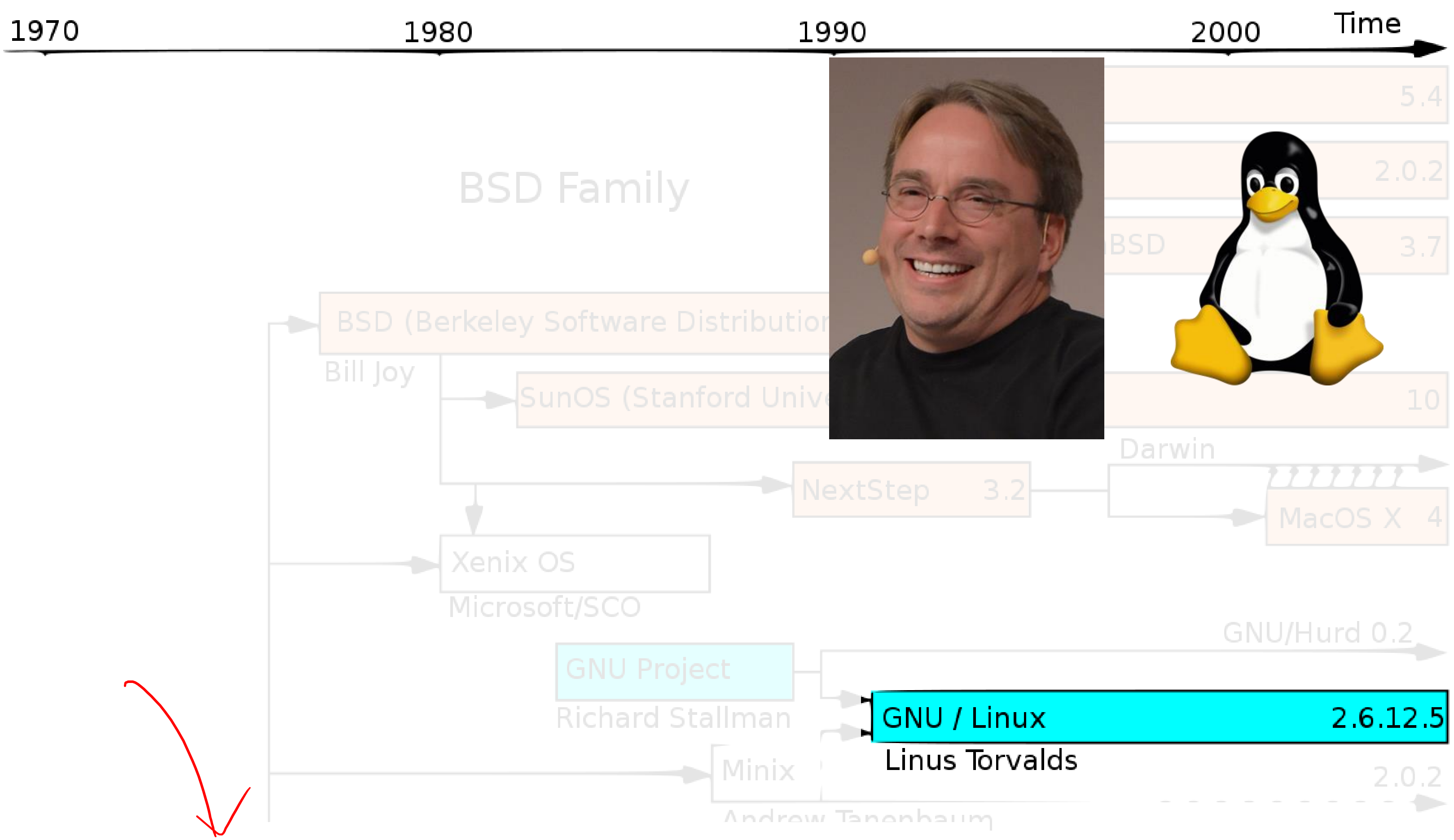
Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

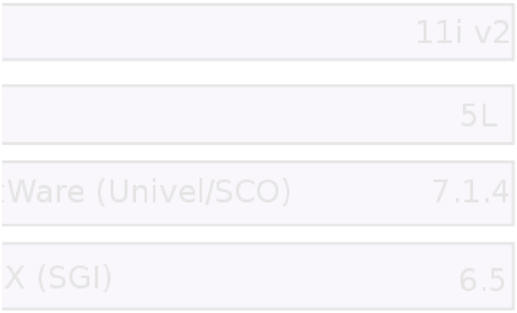
Linus (torv...@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.



Free (libre) Linux Distributions:

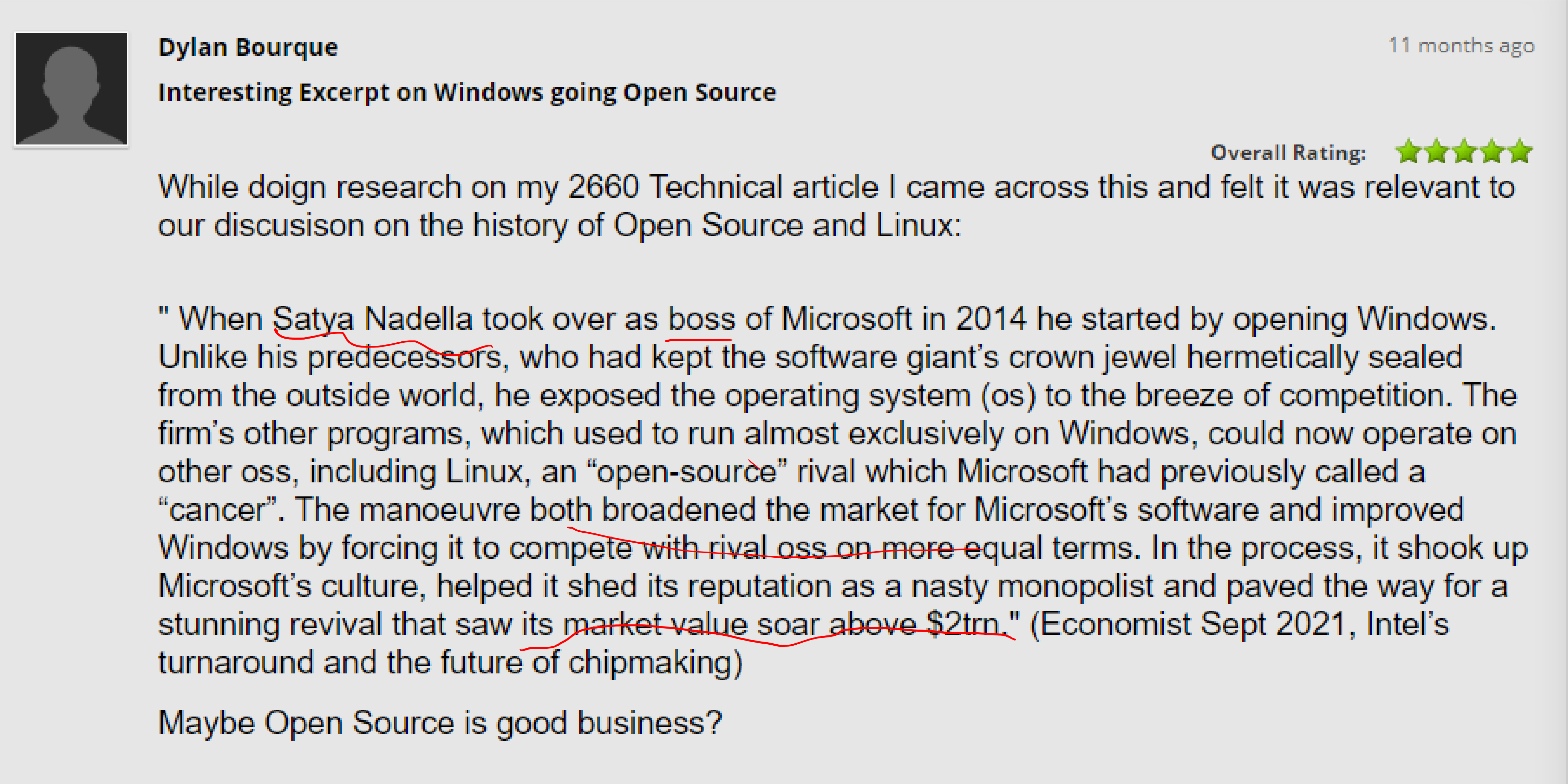
- Debian
- Fedora
- Ubuntu



Commercial Distributions:

- Red Hat Enterprise Linux
- SUSE Linux Enterprise Server.

https://www.youtube.com/watch?v=qrYt4bbEUrU



Dylan Bourque

11 months ago

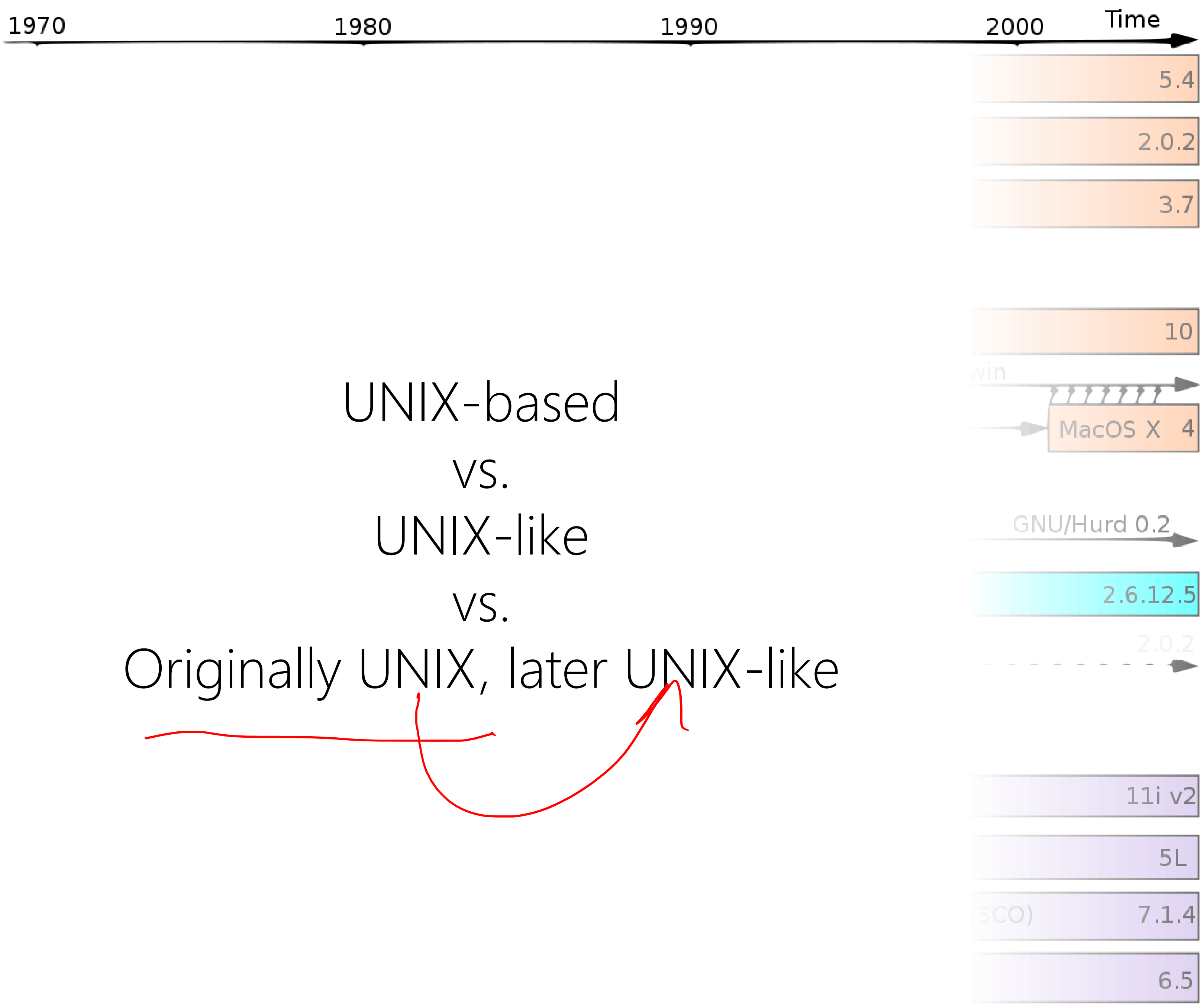
Interesting Excerpt on Windows going Open Source

Overall Rating: ★★★★★

While doign research on my 2660 Technical article I came across this and felt it was relevant to our discusison on the history of Open Source and Linux:

" When Satya Nadella took over as boss of Microsoft in 2014 he started by opening Windows. Unlike his predecessors, who had kept the software giant's crown jewel hermetically sealed from the outside world, he exposed the operating system (os) to the breeze of competition. The firm's other programs, which used to run almost exclusively on Windows, could now operate on other oss, including Linux, an "open-source" rival which Microsoft had previously called a "cancer". The manoeuvre both broadened the market for Microsoft's software and improved Windows by forcing it to compete with rival oss on more equal terms. In the process, it shook up Microsoft's culture, helped it shed its reputation as a nasty monopolist and paved the way for a stunning revival that saw its market value soar above $2trn." (Economist Sept 2021, Intel's turnaround and the future of chipmaking)

Maybe Open Source is good business?



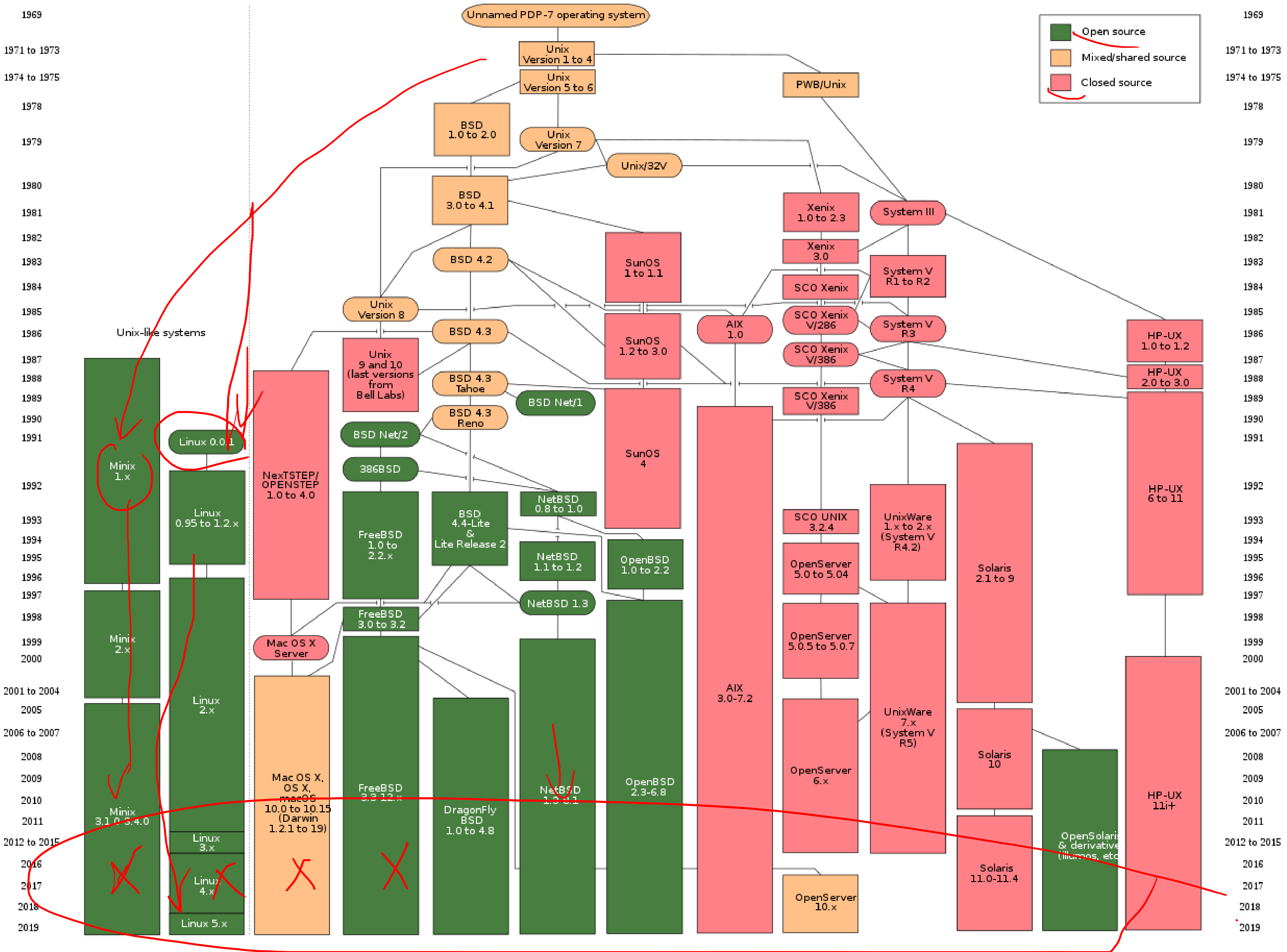
UNIX-based

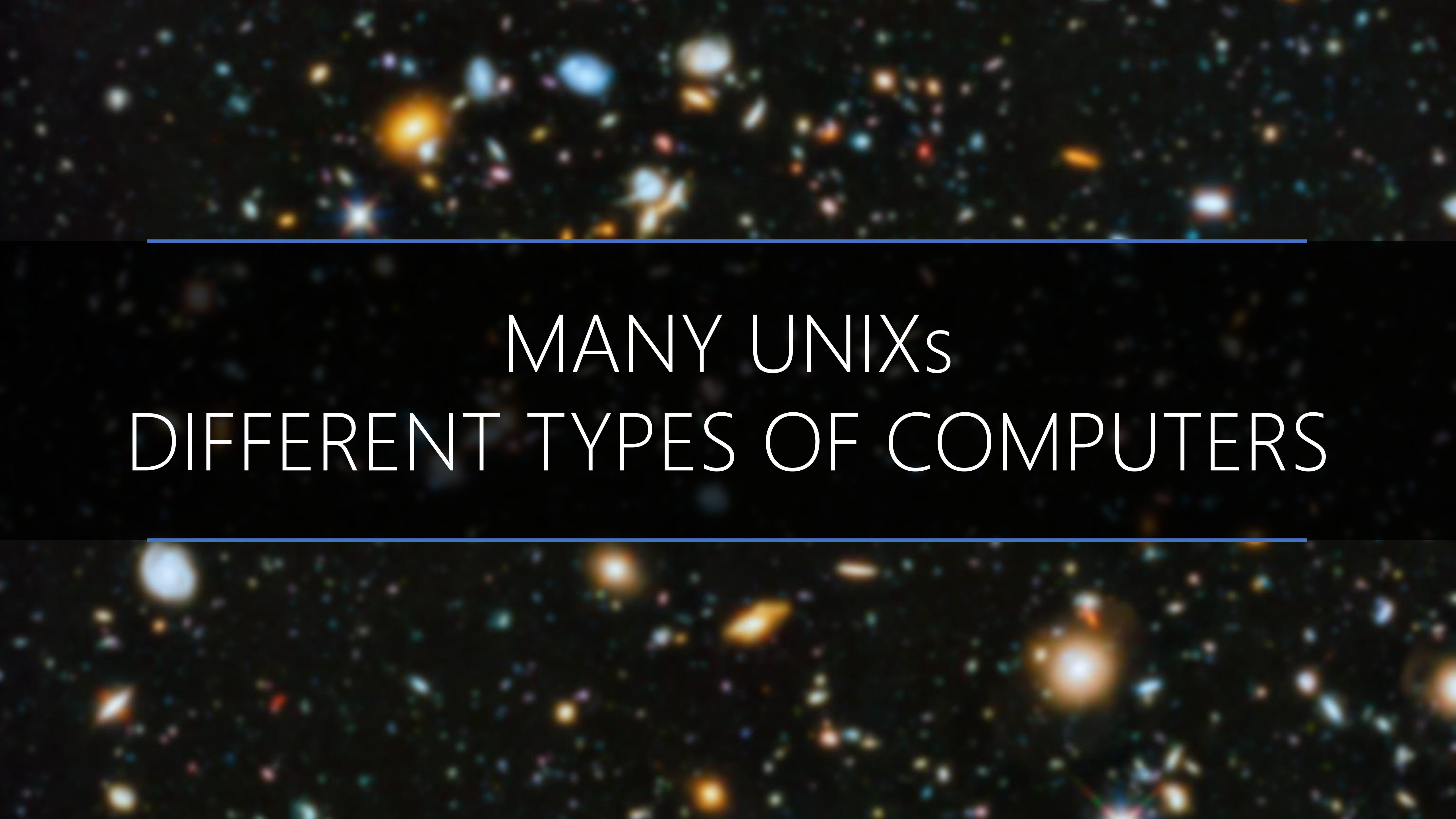
vs.

UNIX-like

vs.

Originally UNIX, later UNIX-like


Unnamed PDP-7 operating system
Open source
Mixed/shared source
Closed source
1969
1971 to 1973
1974 to 1975
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001 to 2004
2005
2006 to 2007
2008
2009
2010
2011
2012 to 2015
2016
2017
2018
2019
Unix Version 1 to 4
Unix Version 5 to 6
PWB/Unix
BSD 1.0 to 2.0
Unix Version 7
Unix/32V
BSD 3.0 to 4.1
Xenix 1.0 to 2.3
System III
Xenix 3.0
BSD 4.2
SunOS 1 to 1.1
System V R1 to R2
SCO Xenix
Unix Version 8
SCO Xenix V/286
System V R3
AIX 1.0
Unix-like systems
BSD 4.3
SunOS 1.2 to 3.0
HP-UX 1.0 to 1.2
Unix 9 and 10 (last versions from Bell Labs)
SCO Xenix V/386
HP-UX 2.0 to 3.0
BSD 4.3 Tahoe
System V R4
BSD Net/1
SCO Xenix V/386
BSD 4.3 Reno
Linux 0.0.1
BSD Net/2
SunOS 4
Minix 1.x
386BSD
NeXTSTEP/ OPENSTEP 1.0 to 4.0
HP-UX 6 to 11
NetBSD 0.8 to 1.0
Linux 0.95 to 1.2.x
BSD 4.4-Lite & Lite Release 2
SCO UNIX 3.2.4
UnixWare 1.x to 2.x (System V R4.2)
FreeBSD 1.0 to 2.2.x
NetBSD 1.1 to 1.2
OpenBSD 1.0 to 2.2
OpenServer 5.0 to 5.04
Solaris 2.1 to 9
NetBSD 1.3
FreeBSD 3.0 to 3.2
Minix 2.x
Mac OS X Server
OpenServer 5.0.5 to 5.0.7
AIX 3.0-7.2
Linux 2.x
UnixWare 7.x (System V R5)
Solaris 10
Mac OS X, OS X, macOS 10.0 to 10.15 (Darwin 1.2.1 to 19)
FreeBSD 3.3-12.x
OpenServer 6.x
NetBSD 1.3-8.1
OpenBSD 2.3-6.8
HP-UX 11i+
DragonFly BSD 1.0 to 4.8
Minix 3.1.0-3.4.0
Linux 3.x
OpenSolaris & derivatives (illumos, etc.)
Solaris 11.0-11.4
Linux 4.x
OpenServer 10.x
Linux 5.x



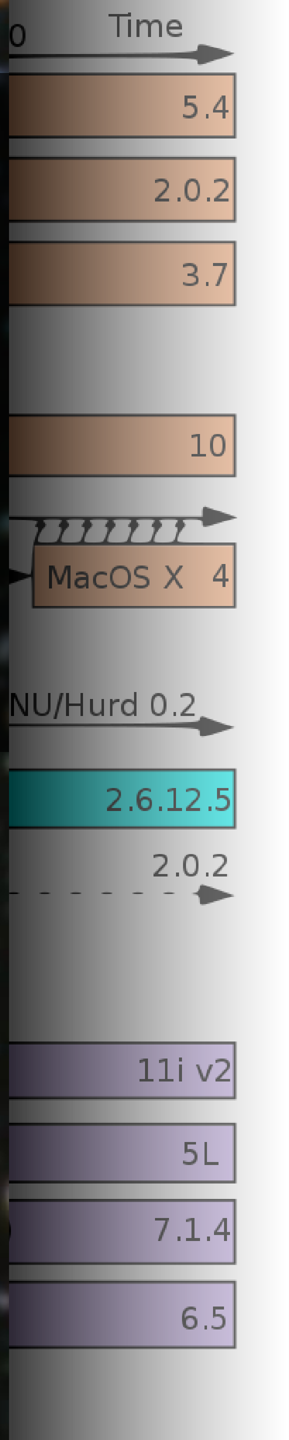
MANY UNIXs
DIFFERENT TYPES OF COMPUTERS

STANDARD

UNIX wars

e.g., ask kernel to allocate memory (System Call)

UNIX:	xxxx(10 byte)
Linux-Debian	yyyy(10 byte)
Linux-Ubuntu	Not Available
MacOS	aaaa(10 byte)
OpenBSD	Not Available
....	



e.g., ask kernel to allocate memory (System Call)

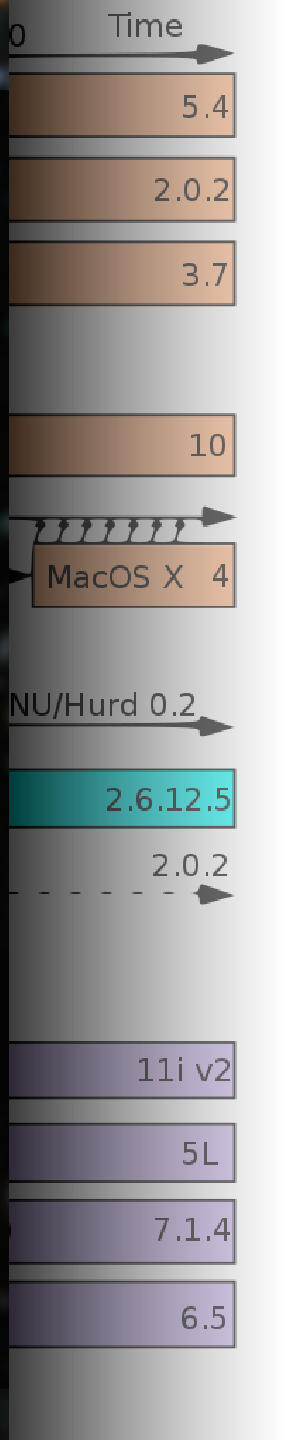
UNIX:	xxxx(10 byte)
Linux-Debian	xxxx(10 byte)
Linux-Ubuntu	xxxx(10 byte)
MacOS	xxxx(10 byte)
OpenBSD	xxxx(10 byte)
….	

A min common-denominator ~~system~~ interface

Same Set of System Calls

Each System Call

- Same input, same result. How is not important!
- 10 byte = 5 byte + 5 byte = 1 byte + 9 byte = 10 byte



The Open Group owns the UNIX trademark and uses the Single UNIX Specification to define the interfaces an implementation must support to call itself a UNIX system. Vendors must file conformance statements, pass test suites to verify conformance, and license the right to use the UNIX trademark.

POSIX

Portable Operating System Interface

a family of standards by the IEEE Computer Society for maintaining compatibility between operating systems

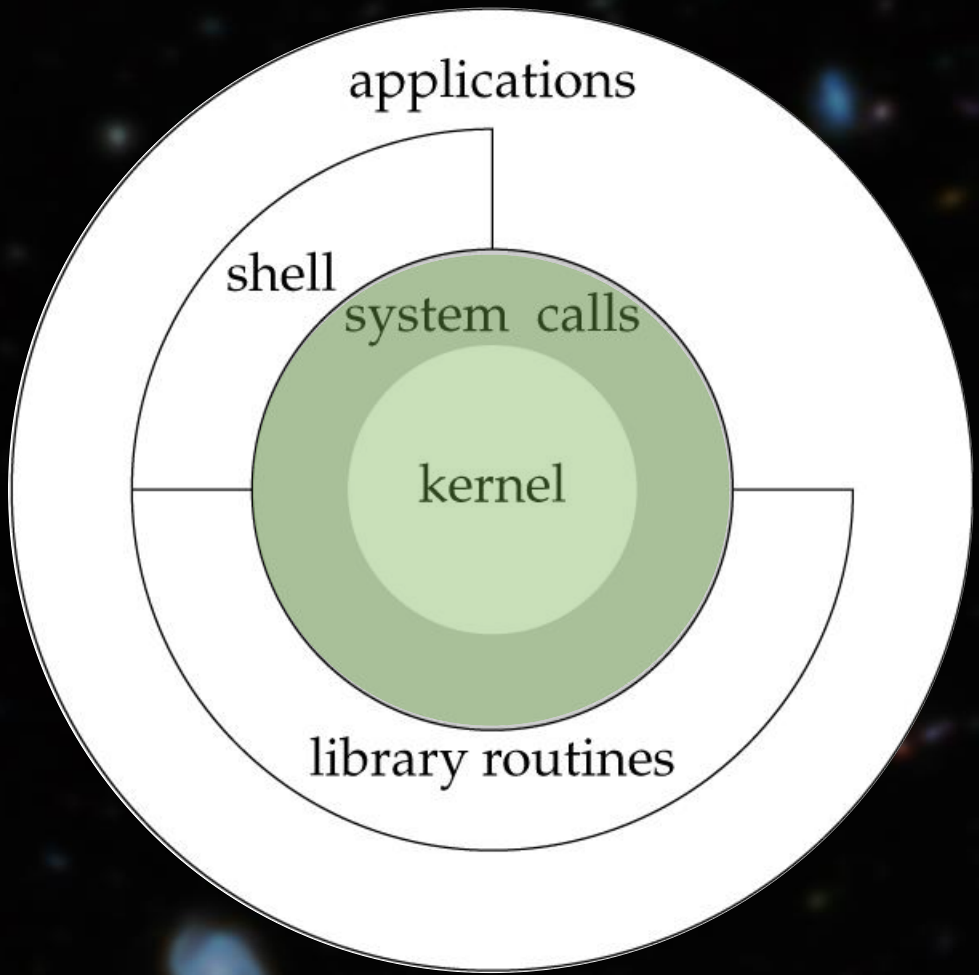
POSIX

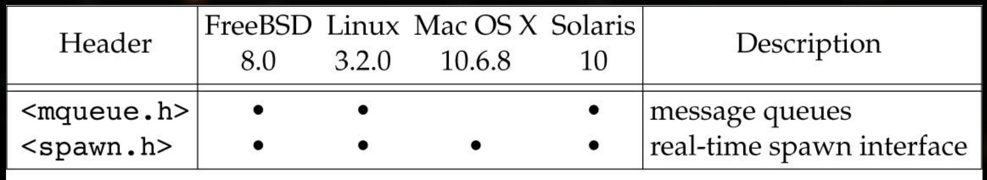
Is UNIX POSIX-compliant? ✗

Is MacOS POSIX-compliant? ✓

Is Linux POSIX-compliant? ✗

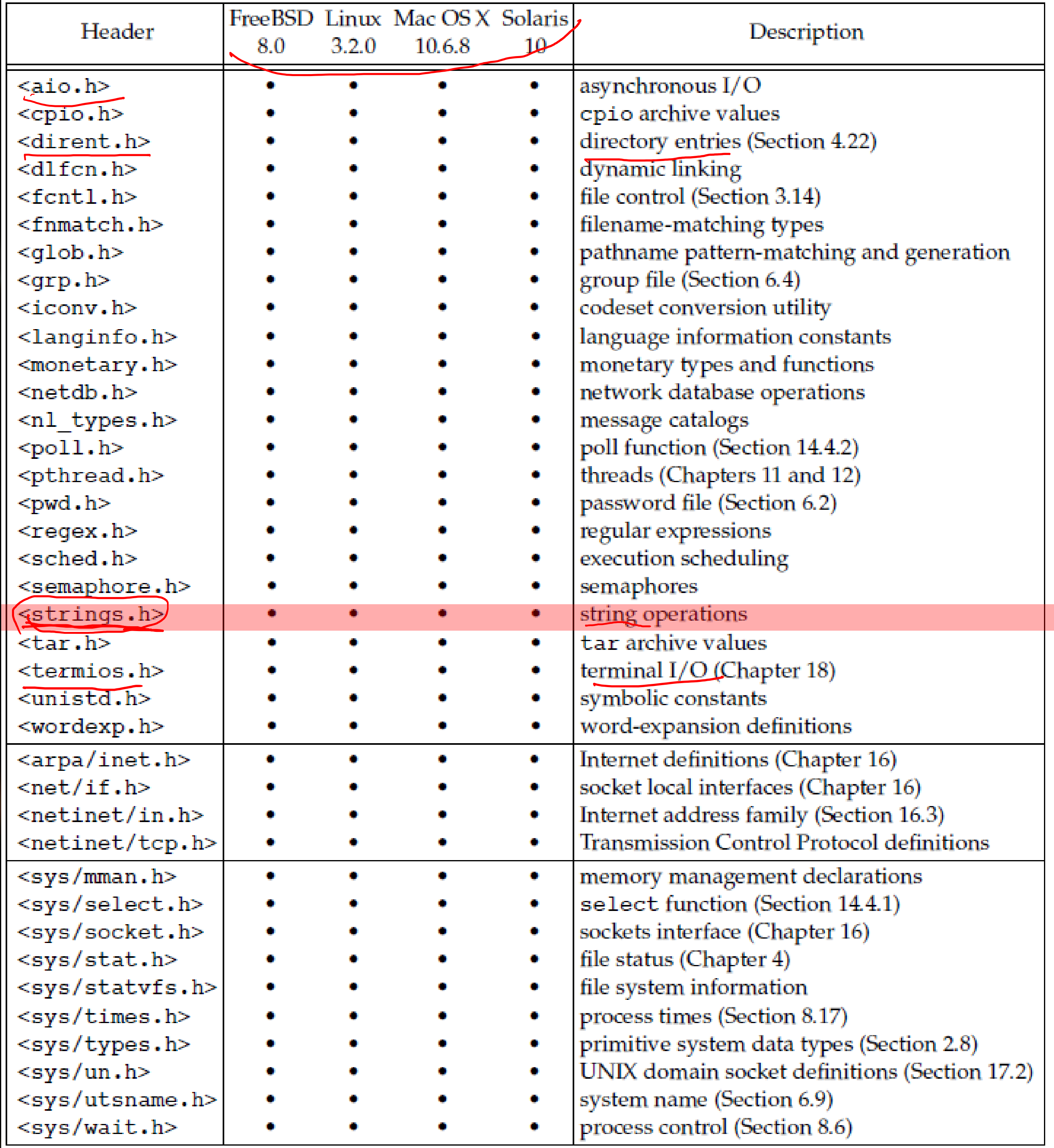
Is Windows10 POSIX-compliant? ✗



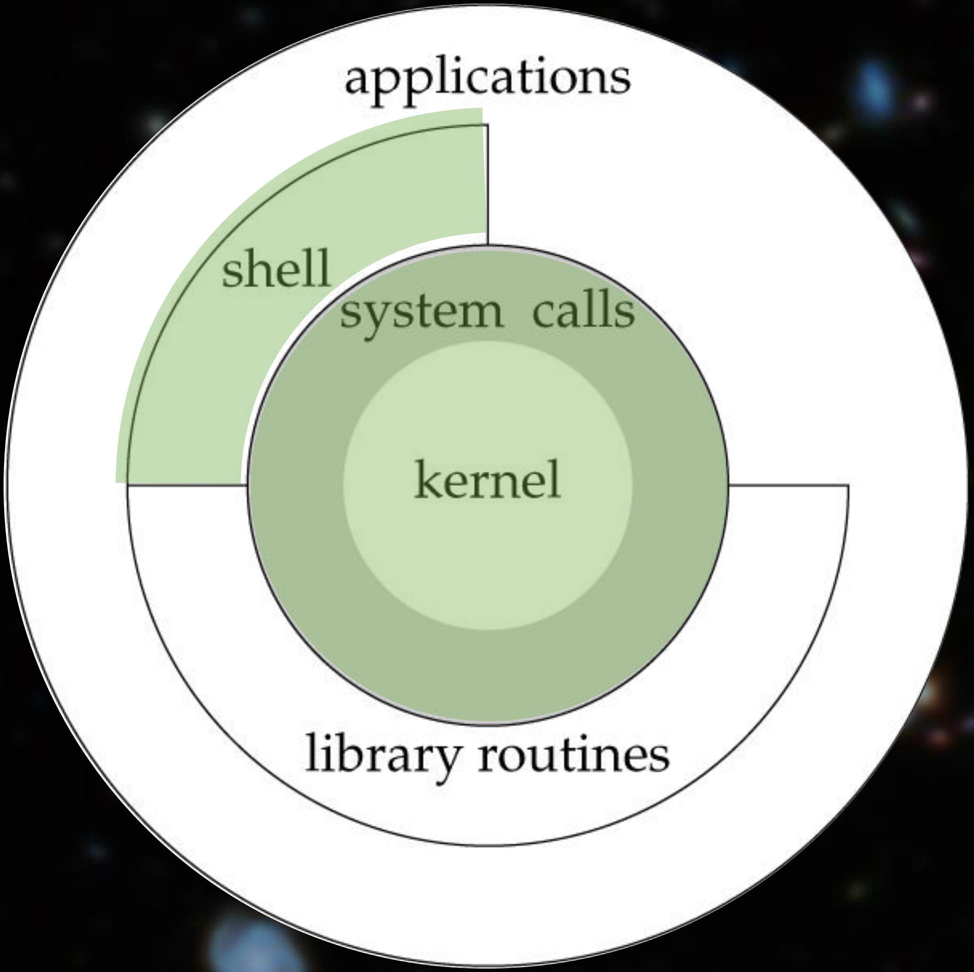


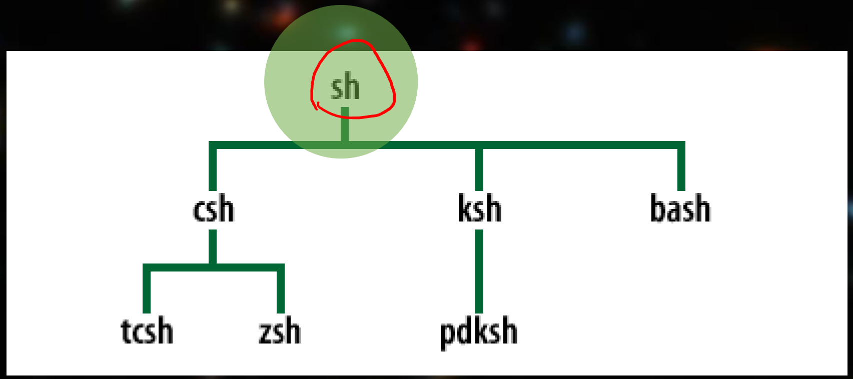
Header	FreeBSD 8.0	Linux 3.2.0	Mac OS X 10.6.8	Solaris 10	Description
`<mqueue.h>`	•	•		•	message queues
`<spawn.h>`	•	•	•	•	real-time spawn interface

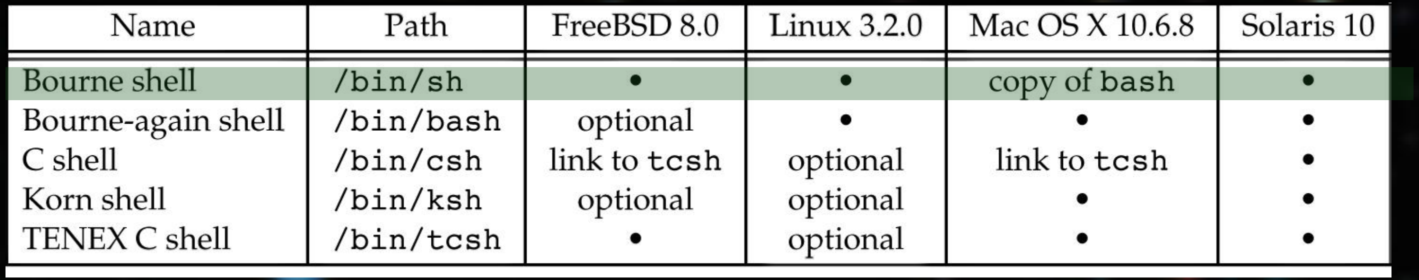
Figure 2.4 Optional headers defined by the POSIX standard



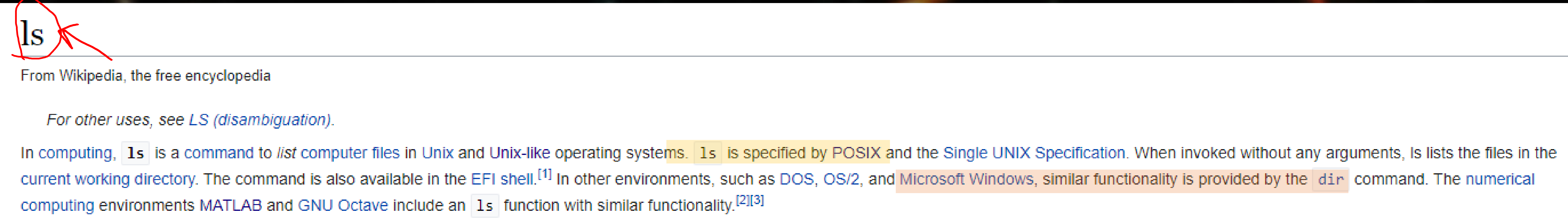
Header	FreeBSD 8.0	Linux 3.2.0	Mac OS X 10.6.8	Solaris 10	Description
`<aio.h>`	•	•	•	•	asynchronous I/O
`<cpio.h>`	•	•	•	•	cpio archive values
`<dirent.h>`	•	•	•	•	directory entries (Section 4.22)
`<dlfcn.h>`	•	•	•	•	dynamic linking
`<fcntl.h>`	•	•	•	•	file control (Section 3.14)
`<fnmatch.h>`	•	•	•	•	filename-matching types
`<glob.h>`	•	•	•	•	pathname pattern-matching and generation
`<grp.h>`	•	•	•	•	group file (Section 6.4)
`<iconv.h>`	•	•	•	•	codeset conversion utility
`<langinfo.h>`	•	•	•	•	language information constants
`<monetary.h>`	•	•	•	•	monetary types and functions
`<netdb.h>`	•	•	•	•	network database operations
`<nl_types.h>`	•	•	•	•	message catalogs
`<poll.h>`	•	•	•	•	poll function (Section 14.4.2)
`<pthread.h>`	•	•	•	•	threads (Chapters 11 and 12)
`<pwd.h>`	•	•	•	•	password file (Section 6.2)
`<regex.h>`	•	•	•	•	regular expressions
`<sched.h>`	•	•	•	•	execution scheduling
`<semaphore.h>`	•	•	•	•	semaphores
`<strings.h>`	•	•	•	•	string operations
`<tar.h>`	•	•	•	•	tar archive values
`<termios.h>`	•	•	•	•	terminal I/O (Chapter 18)
`<unistd.h>`	•	•	•	•	symbolic constants
`<wordexp.h>`	•	•	•	•	word-expansion definitions
`<arpa/inet.h>`	•	•	•	•	Internet definitions (Chapter 16)
`<net/if.h>`	•	•	•	•	socket local interfaces (Chapter 16)
`<netinet/in.h>`	•	•	•	•	Internet address family (Section 16.3)
`<netinet/tcp.h>`	•	•	•	•	Transmission Control Protocol definitions
`<sys/mman.h>`	•	•	•	•	memory management declarations
`<sys/select.h>`	•	•	•	•	select function (Section 14.4.1)
`<sys/socket.h>`	•	•	•	•	sockets interface (Chapter 16)
`<sys/stat.h>`	•	•	•	•	file status (Chapter 4)
`<sys/statvfs.h>`	•	•	•	•	file system information
`<sys/times.h>`	•	•	•	•	process times (Section 8.17)
`<sys/types.h>`	•	•	•	•	primitive system data types (Section 2.8)
`<sys/un.h>`	•	•	•	•	UNIX domain socket definitions (Section 17.2)
`<sys/utsname.h>`	•	•	•	•	system name (Section 6.9)
`<sys/wait.h>`	•	•	•	•	process control (Section 8.6)







Name	Path	FreeBSD 8.0	Linux 3.2.0	Mac OS X 10.6.8	Solaris 10
Bourne shell	/bin/sh	•	•	copy of bash	•
Bourne-again shell	/bin/bash	optional	•	•	•
C shell	/bin/csh	link to tcsh	optional	link to tcsh	•
Korn shell	/bin/ksh	optional	optional	•	•
TENEX C shell	/bin/tcsh	•	optional	•	•

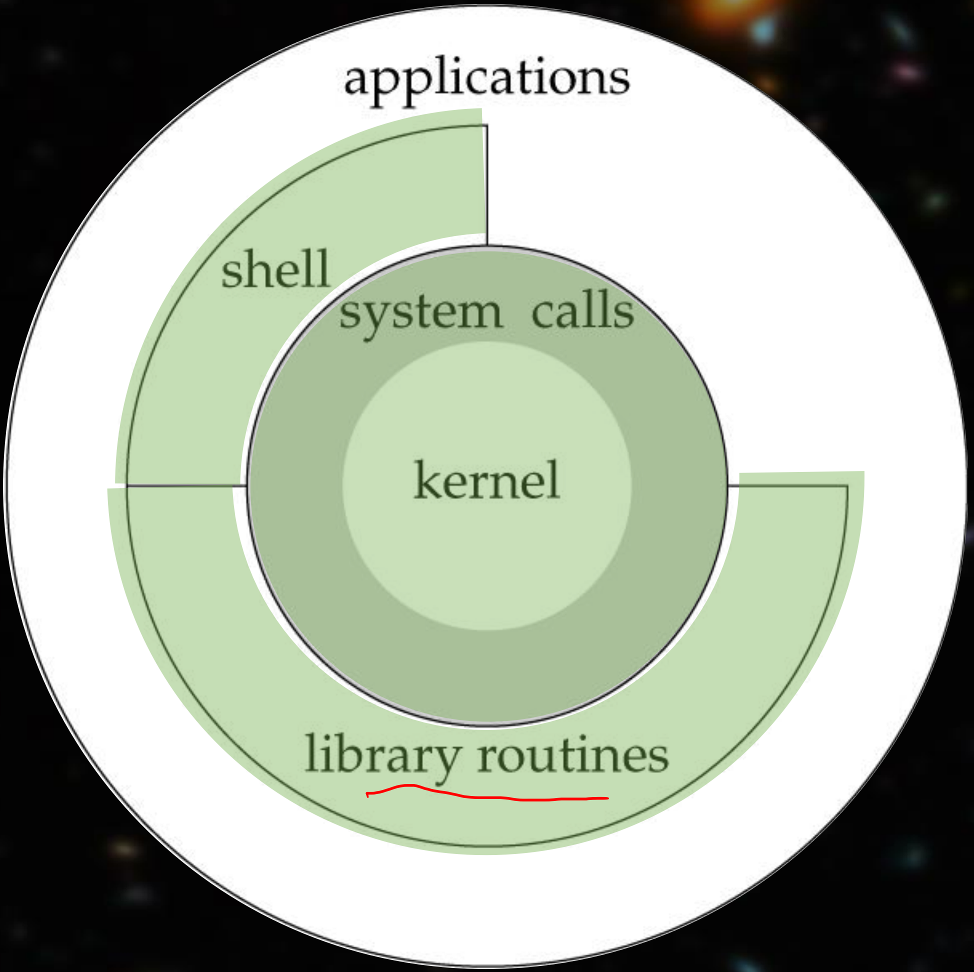


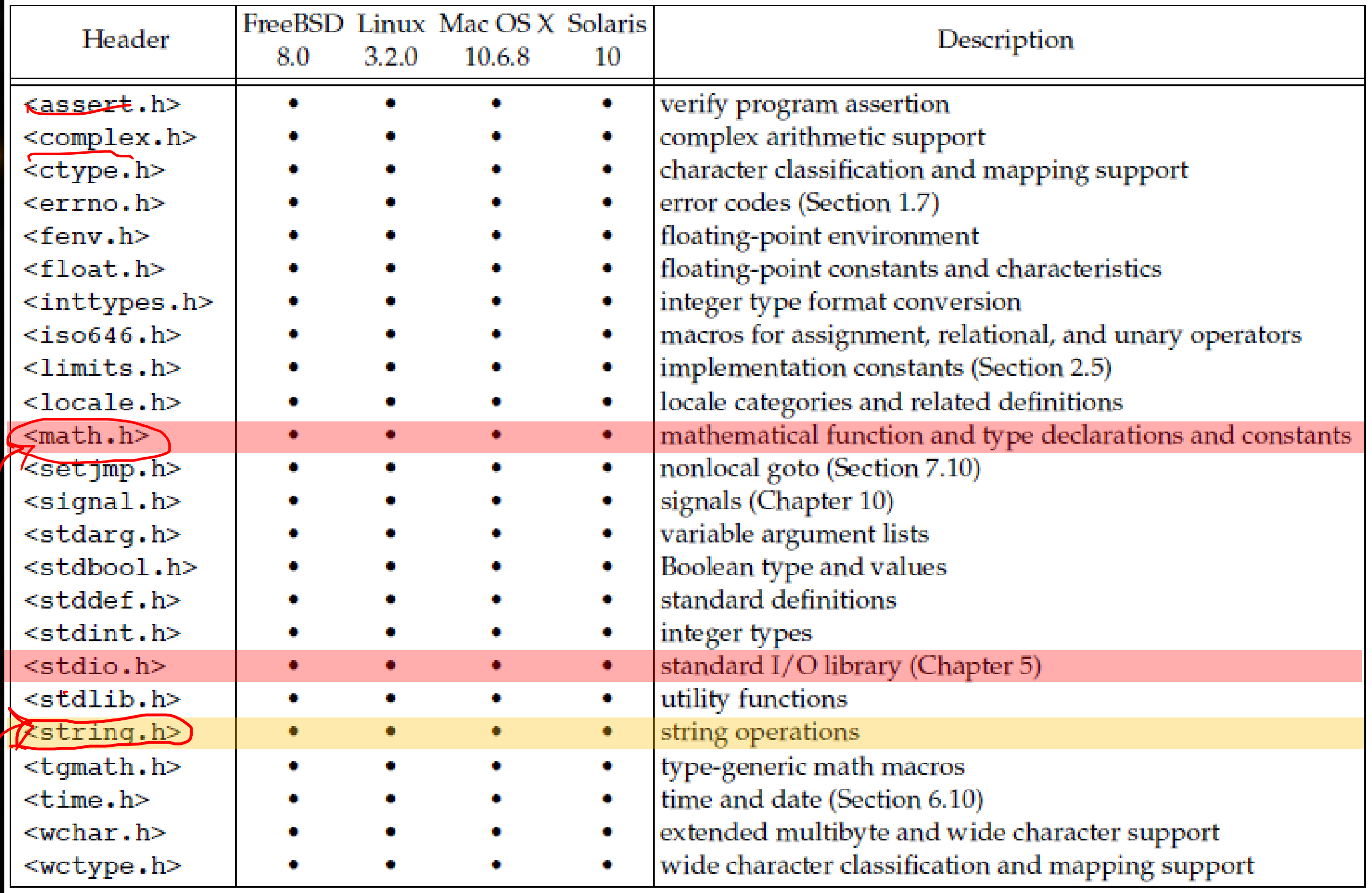
ls

From Wikipedia, the free encyclopedia

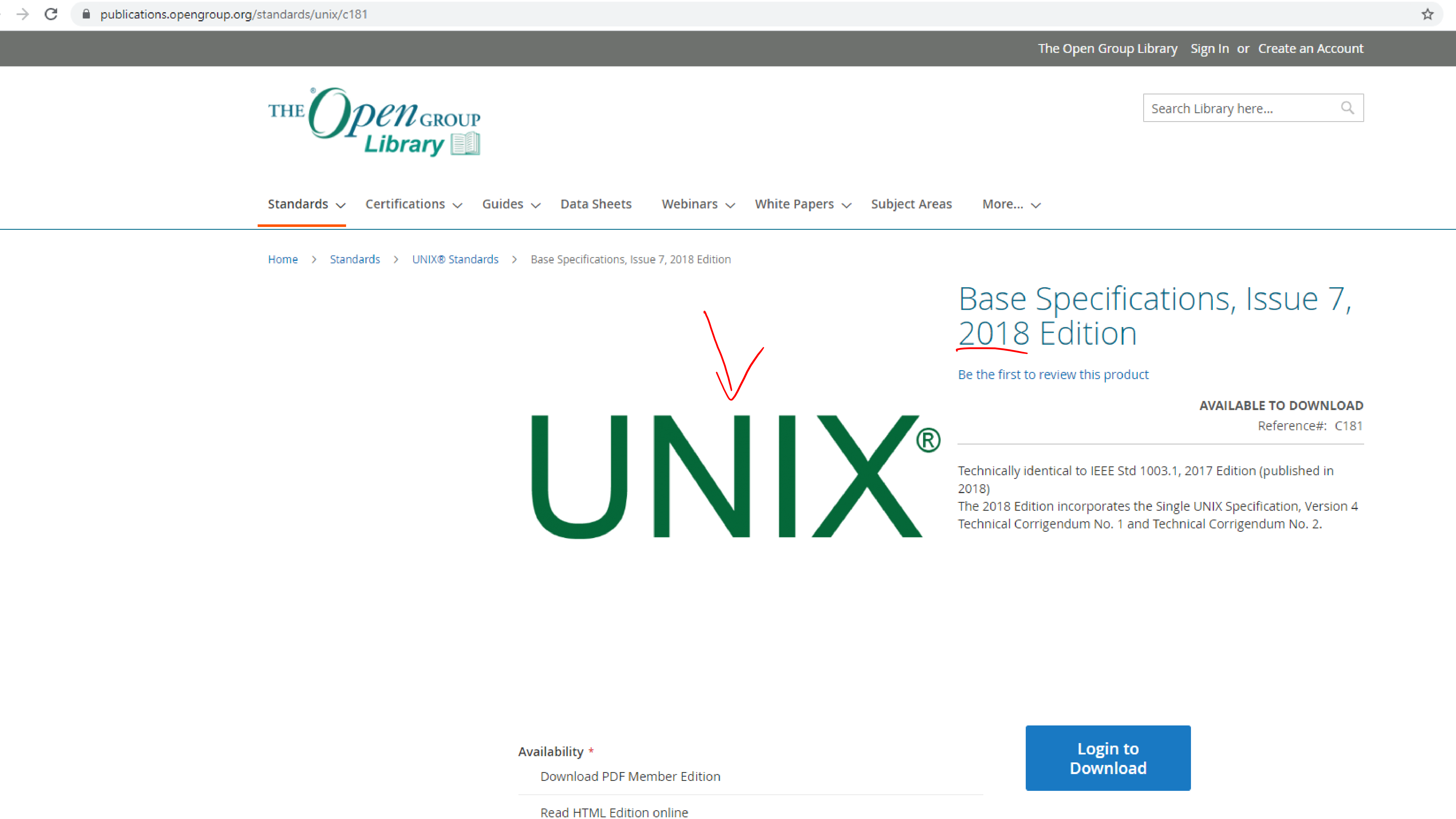
For other uses, see LS (disambiguation).

In computing, `ls` is a command to *list* computer files in Unix and Unix-like operating systems. `ls` is specified by POSIX and the Single UNIX Specification. When invoked without any arguments, ls lists the files in the current working directory. The command is also available in the EFI shell.[1] In other environments, such as DOS, OS/2, and Microsoft Windows, similar functionality is provided by the `dir` command. The numerical computing environments MATLAB and GNU Octave include an `ls` function with similar functionality.[2][3]





Header	FreeBSD 8.0	Linux 3.2.0	Mac OS X 10.6.8	Solaris 10	Description
<assert.h>	•	•	•	•	verify program assertion
<complex.h>	•	•	•	•	complex arithmetic support
<ctype.h>	•	•	•	•	character classification and mapping support
<errno.h>	•	•	•	•	error codes (Section 1.7)
<fenv.h>	•	•	•	•	floating-point environment
<float.h>	•	•	•	•	floating-point constants and characteristics
<inttypes.h>	•	•	•	•	integer type format conversion
<iso646.h>	•	•	•	•	macros for assignment, relational, and unary operators
<limits.h>	•	•	•	•	implementation constants (Section 2.5)
<locale.h>	•	•	•	•	locale categories and related definitions
<math.h>	•	•	•	•	mathematical function and type declarations and constants
<setjmp.h>	•	•	•	•	nonlocal goto (Section 7.10)
<signal.h>	•	•	•	•	signals (Chapter 10)
<stdarg.h>	•	•	•	•	variable argument lists
<stdbool.h>	•	•	•	•	Boolean type and values
<stddef.h>	•	•	•	•	standard definitions
<stdint.h>	•	•	•	•	integer types
<stdio.h>	•	•	•	•	standard I/O library (Chapter 5)
<stdlib.h>	•	•	•	•	utility functions
<string.h>	•	•	•	•	string operations
<tgmath.h>	•	•	•	•	type-generic math macros
<time.h>	•	•	•	•	time and date (Section 6.10)
<wchar.h>	•	•	•	•	extended multibyte and wide character support
<wctype.h>	•	•	•	•	wide character classification and mapping support



publications.opengroup.org/standards/unix/c181

The Open Group Library Sign In or Create an Account

Home > Standards > UNIX® Standards > Base Specifications, Issue 7, 2018 Edition

Base Specifications, Issue 7, 2018 Edition

Be the first to review this product

AVAILABLE TO DOWNLOAD

Reference#: C181

Technically identical to IEEE Std 1003.1, 2017 Edition (published in 2018)
The 2018 Edition incorporates the Single UNIX Specification, Version 4 Technical Corrigendum No. 1 and Technical Corrigendum No. 2.

Availability *

Download PDF Member Edition

Read HTML Edition online

POSIX-compliant printf

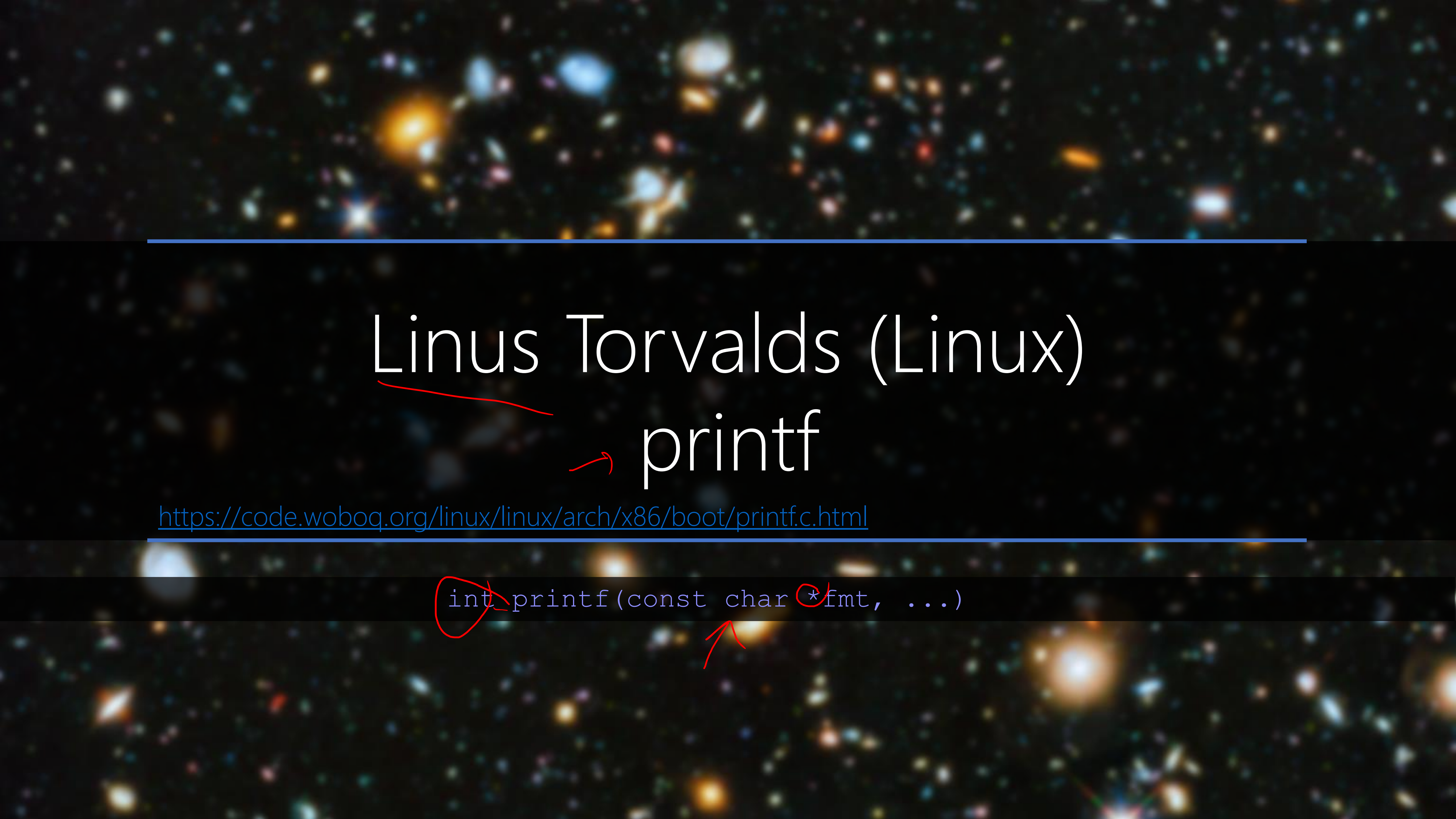
https://pubs.opengroup.org/onlinepubs/9699919799/functions/fprintf.html

```
int printf(const char *restrict format, ...);
```

University of California (BSD) printf

https://unix.superglobalmegacorp.com/Net2/newsrc/stand/printf.c.html

```
void
#if __STDC__
printf(const char *fmt, ...)
#else
printf(fmt /* , va_alist */)
       char *fmt;
#endif
```

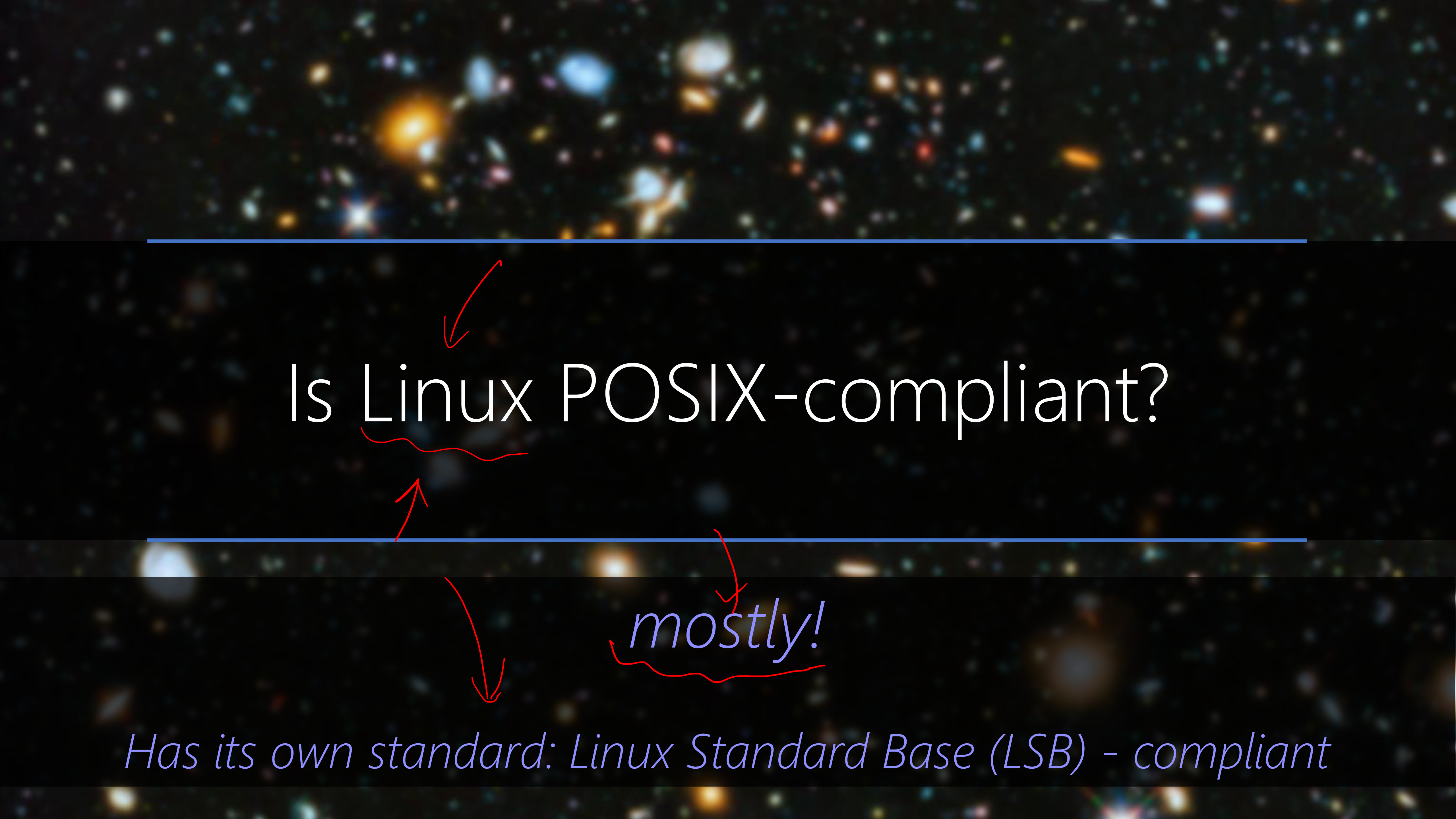


Linus Torvalds (Linux)

printf

https://code.woboq.org/linux/linux/arch/x86/boot/printf.c.html

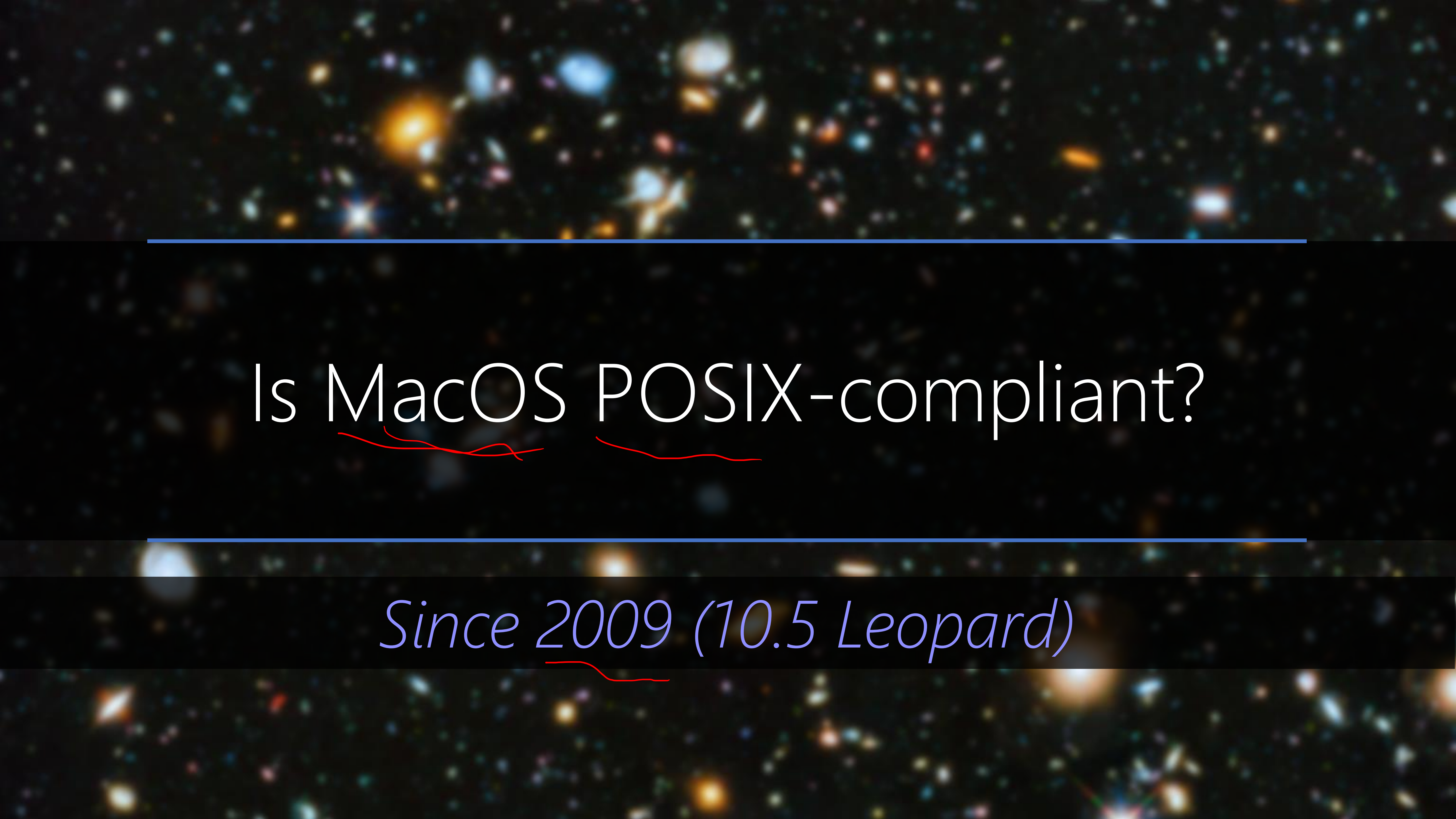
```
int printf(const char *fmt, ...)
```



Is Linux POSIX-compliant?

mostly!

Has its own standard: Linux Standard Base (LSB) - compliant



Is MacOS POSIX-compliant?

Since 2009 (10.5 Leopard)

IRAN