Information Technology Essentials — Lecture 09

Dr. Karim Lounis

Fall 2023

UNIX Vs GNU/Linux Vs Linux

MULTICS

Everything started in the ‘60s when punched cards were still used. A cooperative project, called MULTICS (Multiplexed Information and Computing Service), started in 1964, led by MIT, General Electric, and AT&T Bell Labs, aimed at the development of a time-sharing operating system, Multics, to operate on GE-645 mainframes. The project failed in 1969.



Some employees working on punched-cards computers (on the left), a GE-645 mainframe (in the center), and Denis Ritchie and Ken Thomson (from AT&A Bell Labs) working on a PDP-11/20 (on the right).

Unix

- Ken Thomson, who worked on the project decided to continue working on the project to, at least, create something out of the project aches.
- He formed a team in **1970** and built Unics, which later became Unix.

  The first version of Unics (Unix v1.0) was fully written in assembly and was run on DEC PDP-11/20 minicomputers.
- In 1972, Denis Retchie, a programmer from the team, developed the C programming language.

  In the meantime, the team was progressing and upgrading Unix to v4.0.
- They decided to rewrite Unix in C, creating Unix v5.0 in **1972**.
- At that time, AT&T was forbidden from entering the computer market after some legal issues.

Unix Live Free or Die

- AT&T decided to license the source code of Unix to third commercial and academic parties.

  E.g., UC Berkeley, Microsoft, IBM, DEC, HP and Sun Microsystems.

- In **1977**, the University of California Berkeley was licensed the source code and started developing the system further creating **BSD**.
- In **1980**, Microsoft created Xenix based on Unix v7.0.
- In **1984**, AT&T and Bell Labs got separated and AT&T started selling a commercial Unix version called **System V**.
- From System V and BSD, many versions of Unix were created.

  E.g., HP-UX (1984) and IBM AIX (1986) were both based on System V. Whereas, the ULTRIX (1984) from DEC (for PDP-11 and VAX) and Sun Microsystems Solaris (1981) were based on BSD.

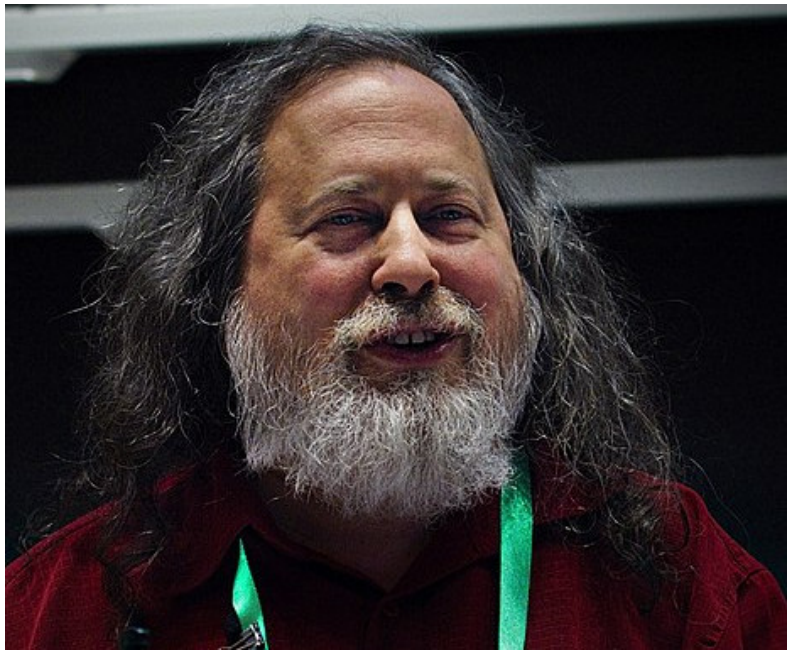
Unix Live Free or Die

- The UC Berkeley started replacing AT&T files with their own files to become separate from AT&T and System V. It managed to publish its source code and create FreeBSD in **1983**.

  E.g., Of course, there was a lawsuit with AT&T and got dropped.

GNU Project

- In **1983**, Richard Stallman founded the GNU foundation (which stands for GNU is Not Unix) with the aim of developing an operating system as a free software replacement for Unix (1984).



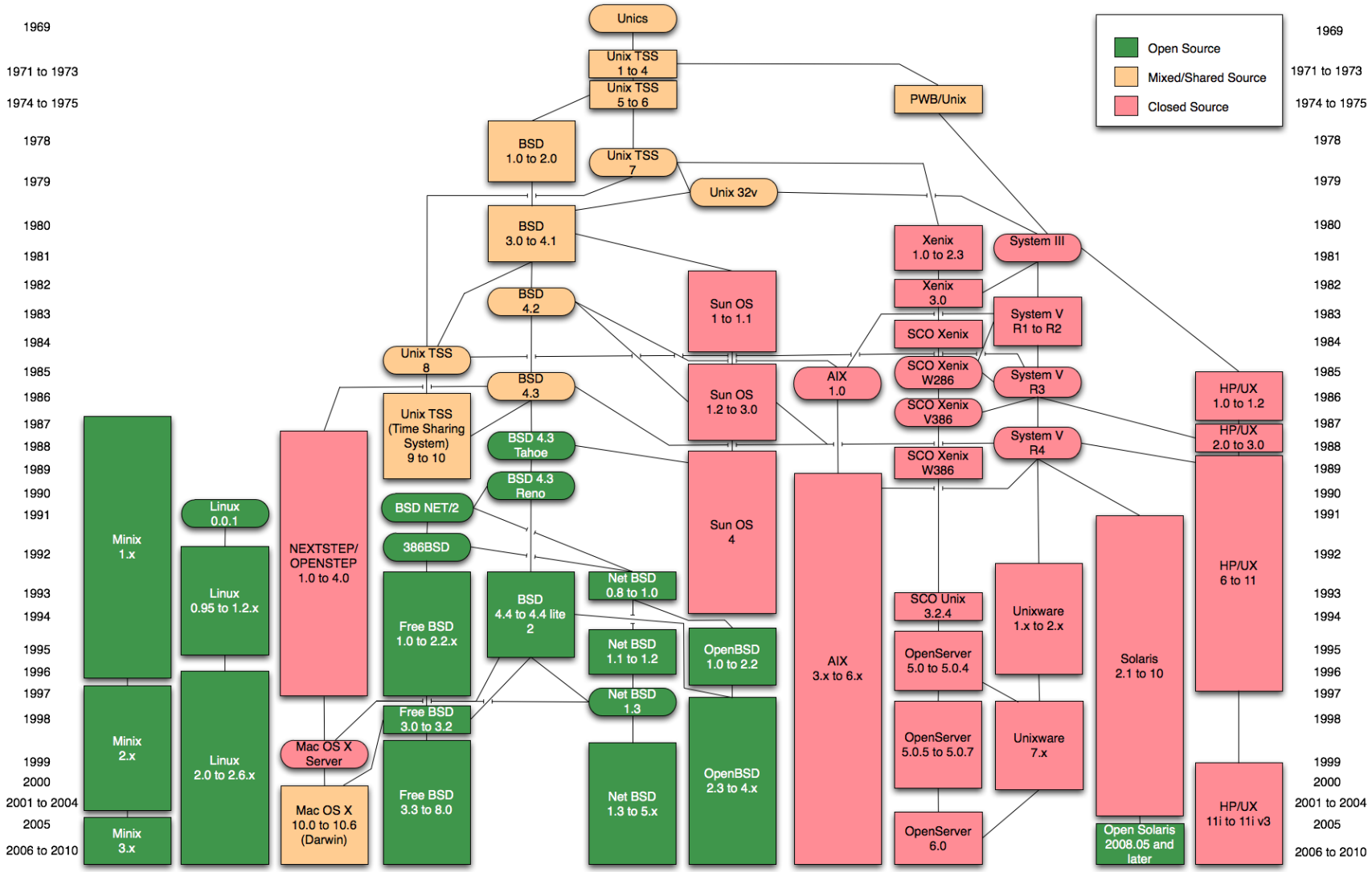
Richard Stallman (Left) and the GNU project logo (right).

GNU Project

- In the early '90s, the GNU system was almost complete but missing an important part, which was the kernel (GNU Hurd was legging behind).
- Few year earlier (1987), somewhere in The Netherlands, Andrew Tanenbaum wrote MINIX, a Unix-like operating system, particularly developed as an educational tool to teach operating systems and get rid of AT&T licence issues.
- At that time, in Finland, Linus Torvald, a student, who used MINIX and knew about Unix wanted to overcome the deficiencies of MINIX and wrote his own operating system (with a kernel called **Linux**) which he wanted to make public.

In **1991**, he released Linux under GPL (GNU General Public Licence). The combination of Linux kernel and the incomplete GNU system made a completely free operating system, called GNU/Linux operating system.

Unix Consequences



GNU GPL

The GNU General Public License (GNU GPL) is a series of widely used free software licenses that guarantee end users the four freedoms to run, study, share, and modify the software. These GPL series are all copyleft licenses, which means that any derivative work must be distributed under the same or equivalent license terms.

Linux Kernel

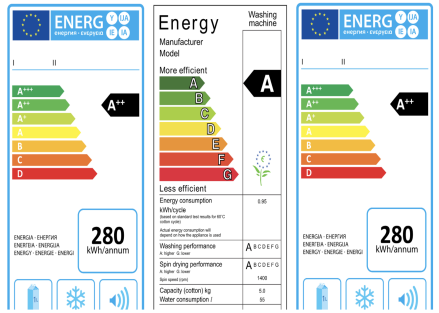
- The Linux kernel is not the Unix kernel. It does not use Unix source code but uses Unix philosophy and architecture.
- Many operating systems nowadays use the Linux kernel. E.g., the Android operating system uses the Linux kernel. Also, 60% of servers outside use Linux. It also runs on embedded systems, including routers, smart home devices, video game consoles, televisions (e.g., Samsung and LG Smart TVs), automobiles (e.g., Tesla, Audi, Mercedes-Benz, Hyundai, and Toyota), and spacecraft (e.g., Falcon 9 rocket).
- There are thousands of GNU/Linux distributions: Red Hat, Fedora, Mandriva, Arch, Ubuntu (spyware), Debian, Backtrack, Kali, Parrot, Linux Mint, Yellow dog, and Kubuntu, these are just distributions (different desktops, . . .) but the same Linux Kernel.

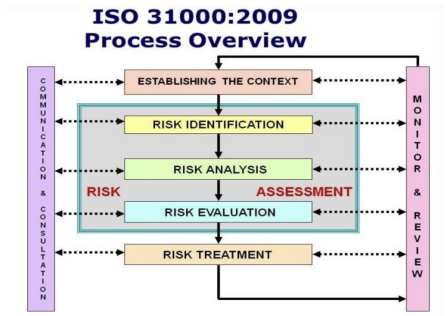
Standards and Interoperability

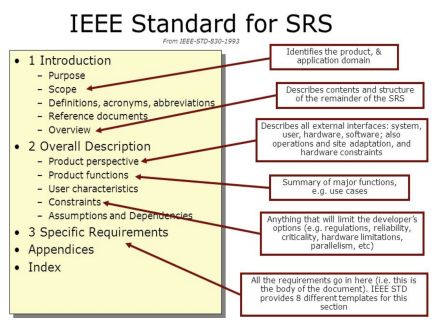
What is a standard and why we should care about?

Standardization

Below are some examples of standards from various fields:

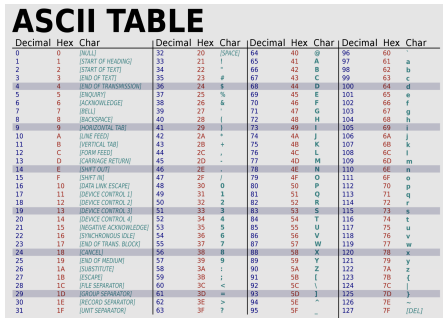


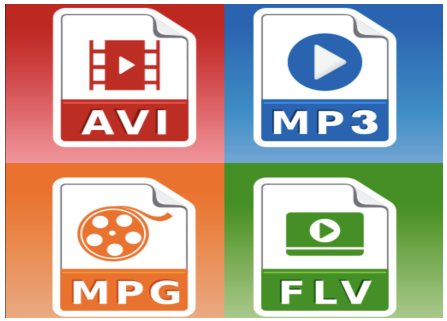


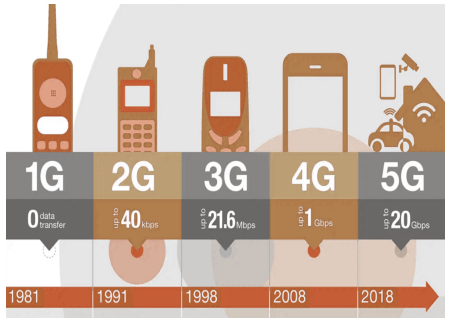


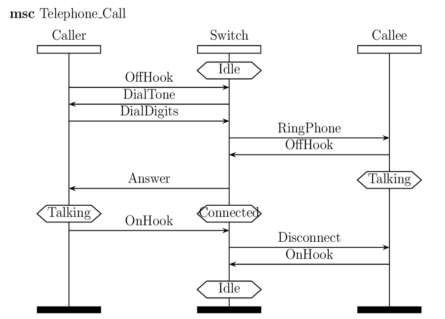
Standardization

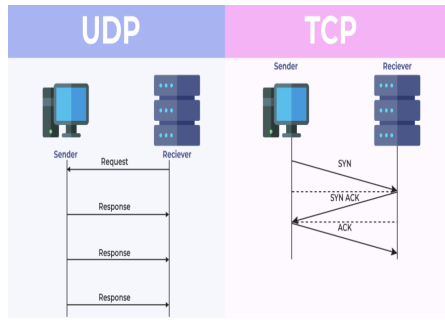
Below are some examples of standards from various fields:











Standardization

Definition

Standardization. Is the process of establishing and following a set of rules, guidlines, practices, and/or specifications in various fields to ensure simplicity, consistency, compatibility, interoperability, quality, accross different products, services, or processes.

For industries:

- Cost of design and development is reduced.
- Expanding customer base and reach.
- Consistency, efficiency, and quality assurance.

For the customers:

- Guarantee compatibility, interoperability, and interchangeability.
- Manufacturer independance and wider selection.
- Safety and Reliability (customer trust).

Standards

Standards are created by either consortiums, national organizations, or international organizations.

- **Consortiums.** It an entrity that groups various manufacturers:
  - EIA (Electronic Industries Association), now merged with TIA (TelecommunicationsIndustry Association). It made RS-232, RS-485, and others.
  - ECMA (European Computer Manufacturer Association), who did ECMA-262 (ECMAScript) & ECMA-404 (TheJSON Data Interchange Format).
  - SIG (Bluetooth Special Interest Group), responsible for Bluetooth technology standards.
  - Wi-Fi Alliance, responsible for developing and promoting Wi-Fi technology standards.
  - OpenAI, responsible for developing and promoting AI standards.
  - USB-IF (Universal Serial Bus Implementers Forum): USB standards.
  - W3C (World Wide Web Consortuim), responsible for the Web.

Standards

Standards are created by either consortiums, national organizations, or international organizations.

- **National Organizations.** These are governmental bodies:
  - IANOR (Institut Algérien de Normalisation), makes standards in various-sectors in Algeria. It seems that ther is a catalogue that you can access https://www.ianor.dz/normalisation/normes-ctn/.
  - DIN (Deutsches Institute für Normung) makes standards in Germany. It made the DIN connectors, DIN 1988, DIN 50018, etc.
  - BSI (British Standards Institution) makes standards in the United Kingdom, e.g., BS 5839, BS EN 206-1, etc.
  - AFNOR (Association Française de Normalization) makes standards in France, e.g., NF Z 67-147, NF D 27-404, and NF EN 206-1.
  - JISC (Japanese Industrial Standards Committee), makes standards in Japan. E.g., JIS D 1601 (automotive safety glass), JIS L 1902 (Children clothings), JIS X 8341-3 (web for people with disabilities), etc.

Standards

Standards are created by either consortiums, national organizations, or international organizations.

- **International Organizations.** These are international bodies:
  - ISO (International Organization for Standardization) develops and publishes international standards across various industries and sectors. E.g., ISO25010 (Software quality), ISO 20000 (IT services), etc.
  - ITU (International Telecommunication Union), responsible for coordinatingand developing international telecommunications standards and regulations.
  - IEEE (Institute of Electrical and Electronics Engineers), e.g., IEEE 802.3, IEEE802.11, IEEE 1394, IEEE 754, etc.
  - IETF (Internet Engineering Task Force) makes standards related to the Internet. Their standards are published in documents, called RFC (RequestFor Comments). E.g., RFC 5321 (SMTP), RFC 1035 (DNS), RFC791 (IPv4), etc (most RFCs become Internet standards).

Type of Standards

There are three types of standards:

1. **De Jure Standards.** A.k.a., formal standards. These are established by recognized standardization organizations and bodies.

   E.g., ISO/IEC 27001, IEEE 802.11, and ITU-T X.509

2. **De Facto Standards.** Come as a consequence of flooding the market or imposing a product in the market (1st product before the competitors), even though they may not have gone through a formal standardization process — could be a globalized habit.

   E.g., HTTP, USB, PDF, Bluetooth, Emojis, etc

3. **Industry Standards.** They are specific to a particular industry or sector. It can be one of the two standards mentioned above.

   E.g., UL 60950-1 (Information Technology Equipment Safety), FSSC22000 (Food Safety Management), etc

Disadvantages of Standards

While standards play a crucial role in promoting compatibility, interoperability, consistency, and etc, there are also some potential disadvantages associated with standards:

- Innovation limitation.
- Compatibility issues due to misinterpretation.
- Bureaucracy.
- Time and cost and resistance to changes.
- Limited customization.
- Outdated standards.

- End.