
Unix i GNU/Linux

Índex

1. Història d'Unix	1
1.1 Els orígens (1969)	1
1.2 La reescriptura en C (1972-1973)	1
1.3 Expansió acadèmica i les "guerres Unix"	1
1.4 El projecte GNU (1983)	2
1.5 Linus Torvalds i el nucli Linux (1991)	2
1.6 Línia temporal resumida	2
2. La filosofia Unix	3
2.1 Fer una cosa i fer-la bé	3
2.2 Programes que es comuniquen entre ells	3
2.3 Tot és un fitxer	3
2.4 Preferir el text pla	3
2.5 Desenvolupament ràpid de prototips	4
2.6 Portabilitat per sobre de l'eficiència	4
3. Els protagonistes	4
3.1 Dennis Ritchie (1941-2011)	4
3.2 Richard Stallman (1953-)	5
3.3 Linus Torvalds (1969-)	6
4. La llicència GPL	7
5. Característiques de GNU/Linux	7
5.1 Programari lliure i de codi obert	7
5.2 Multiusuari i multitasca real	7
5.3 Multiplataforma	8
5.4 Model de permisos i seguretat	8
5.5 Gran varietat de distribucions	8
5.6 Gestió de paquets centralitzada	8
5.7 Estabilitat i rendiment	8
5.8 Personalització i flexibilitat	8
5.9 Línia d'ordres potent	9
6. Resum	9
7. Referències	9
Sobre la història d'Unix	9
Sobre la filosofia Unix	9
Sobre GNU i la GPL	9
Sobre el kernel Linux	10
Sobre les distribucions de GNU/Linux	10
Sobre els protagonistes	10

Aquest document repassa els orígens d'Unix, la filosofia que va marcar el disseny dels sistemes operatius tipus Unix, i les característiques pròpies de GNU/Linux com a sistema operatiu lliure inspirat en aquell llegat.

1. Història d'Unix

1.1 Els orígens (1969)

Unix neix als Bell Labs (AT&T) l'any 1969, de la mà de **Ken Thompson** i **Dennis Ritchie**, com a evolució (i simplificació) d'un projecte anterior molt més ambiciós i complex anomenat **Multics** (Multiplexed Information and Computing Service), desenvolupat conjuntament per MIT, General Electric i Bell Labs.

Quan Bell Labs es retira del projecte Multics per considerar-lo massa costós i complicat, Thompson aprofita un antic miniordinador PDP-7 per crear, en el seu temps lliure, un sistema operatiu més senzill. El nom **Unix** sorgeix com un joc de paraules respecte a Multics: on Multics pretenia fer “moltes coses”, Unix es plantejava fer-ne “una sola, però ben feta”.

1.2 La reescriptura en C (1972-1973)

Inicialment, Unix estava escrit en llenguatge ensamblador, lligat al maquinari del PDP-7 i, després, del PDP-11. Aquest fet limitava molt la seva portabilitat.

Dennis Ritchie desenvolupa el llenguatge **C** precisament amb l'objectiu de reescriure Unix en un llenguatge d'alt nivell. Cap al 1973, la major part del nucli del sistema ja estava escrit en C. Aquesta decisió va ser clau: en separar el sistema operatiu del maquinari concret, Unix es va poder portar a màquines molt diferents amb relativa facilitat, cosa que en aquella època era excepcional.

1.3 Expansió acadèmica i les “guerres Unix”

Durant la dècada de 1970, AT&T no tenia permès comercialitzar programari (per restriccions legals derivades del seu monopoli telefònic), de manera que Unix es va distribuir lliurement, amb el codi font inclòs, a universitats i centres de recerca. Això va generar un ecosistema molt actiu de desenvolupament, amb la **Universitat de Califòrnia a Berkeley** com a nucli destacat, que va donar lloc a la branca **BSD** (Berkeley Software Distribution).

A partir dels anys 80, quan es va permetre a AT&T comercialitzar Unix, van sorgir múltiples versions propietàries (System V, HP-UX, AIX, SunOS/Solaris, Xenix...) que es van anar diferenciant, donant lloc a les anomenades “guerres Unix”: una fragmentació de l'estàndard que en va dificultar la interoperabilitat i que els intents de normalització posteriors (com **POSIX**) van intentar pal·liar.

1.4 El projecte GNU (1983)

El 1983, **Richard Stallman**, llavors investigador al MIT, anuncia el projecte **GNU** (acrònim recursiu de "GNU's Not Unix"), amb l'objectiu de crear un sistema operatiu complet, compatible amb Unix, però compost íntegrament de programari lliure.

Stallman fonamenta el projecte en quatre llibertats bàsiques per a l'usuari del programari: executar-lo amb qualsevol finalitat, estudiar-ne el funcionament, redistribuir-ne còpies, i modificar-lo i distribuir les modificacions. Per protegir legalment aquestes llibertats crea la llicència **GPL** (GNU General Public License).

Durant els anys 80, el projecte GNU desenvolupa eines fonamentals: el compilador **GCC**, l'editor **Emacs**, l'interpret d'ordres **Bash**, i moltes utilitats Unix reescrites des de zero (ls, grep, sed, etc.). Cap a finals de la dècada, GNU té gairebé un sistema operatiu complet... excepte el nucli.

1.5 Linus Torvalds i el nucli Linux (1991)

El 1991, un estudiant finlandès anomenat **Linus Torvalds** comença a desenvolupar, com a projecte personal, un nucli de sistema operatiu per al seu propi PC, inspirat en **Minix** (un Unix acadèmic creat per Andrew Tanenbaum). L'anuncia en un missatge a Usenet on, modestament, diu que es tracta "només" d'un hobby i que no serà "gran ni professional" com GNU.

Aquest nucli, anomenat **Linux**, es publica sota llicència GPL i s'acaba combinant amb les eines del projecte GNU, que ja existien, però els faltava precisament un nucli lliure. D'aquesta unió neix el que avui coneixem com a **GNU/Linux**: un sistema operatiu completament lliure compost del nucli Linux i les eines i utilitats de GNU.

1.6 Línia temporal resumida

Any	Fet
1969	Ken Thompson i Dennis Ritchie creen Unix als Bell Labs
1972-73	Reescriptura d'Unix en llenguatge C
1977	Primera versió de BSD (Berkeley)
1983	Richard Stallman anuncia el projecte GNU
1985	Es crea la Free Software Foundation
1989	Primera versió de la llicència GPL
1991	Linus Torvalds anuncia el nucli Linux
1992	Linux es llicencia sota GPL i s'integra amb les eines GNU

2. La filosofia Unix

Més enllà del codi, Unix va deixar un conjunt de principis de disseny que han influït profundament en la informàtica posterior. Aquesta filosofia es resumeix sovint en algunes màximes, popularitzades especialment per Doug McIlroy (un dels creadors del concepte de *pipe*):

2.1 Fer una cosa i fer-la bé

Cada programa hauria de fer **una sola tasca** i fer-la correctament, en lloc d'acumular múltiples funcionalitats en una sola eina. Per exemple, `grep` només cerca patrons de text; no ordena, no compta paraules ni edita fitxers. Per a cada cosa hi ha una altra eina especialitzada (`sort`, `wc`, `sed`...).

2.2 Programes que es comuniquen entre ells

La sortida d'un programa hauria de poder ser l'entrada d'un altre, sense formats rígids ni innecessaris. Això es materialitza en el concepte de **canonada** (*pipe*, representat amb el símbol `|`), que permet encadenar ordres senzilles per construir solucions complexes:

```
cat access.log | grep "404" | awk '{print $1}' | sort | uniq -c | sort  
↪ -nr
```

Aquesta única línia combina cinc eines petites per obtenir, per exemple, un rànquing d'IP que han generat errors 404 en un registre web.

2.3 Tot és un fitxer

Una de les abstraccions més potents d'Unix és tractar la majoria de recursos del sistema ---fitxers normals, directoris, dispositius físics (discos, terminals, impressores), processos (a través de `/proc`)--- com si fossin **fitxers**, accessibles amb les mateixes operacions bàsiques (obrir, llegir, escriure, tancar). Això simplifica enormement la programació de sistemes i l'administració.

2.4 Preferir el text pla

Unix prioritza els formats de text llegibles per humans per emmagatzemar configuració i dades, en lloc de formats binaris opacs. Això permet utilitzar les mateixes eines genèriques (`grep`, `sed`, `awk`, editors de text) per inspeccionar i modificar gairebé qualsevol cosa al sistema, des de la configuració de xarxa fins als registres (logs).

2.5 Desenvolupament ràpid de prototips

La filosofia Unix valora poder construir una primera versió funcional ràpidament, encara que sigui imperfecta, per després iterar i millorar-la, en lloc de dissenyar-ho tot de manera exhaustiva abans de programar res.

2.6 Portabilitat per sobre de l'eficiència

Es prioritza que el programari funcioni en el màxim de plataformes possible (gràcies, en bona part, a estar escrit en C) per sobre d'aprofitar al màxim les particularitats d'un maquinari concret.

3. Els protagonistes

3.1 Dennis Ritchie (1941-2011)

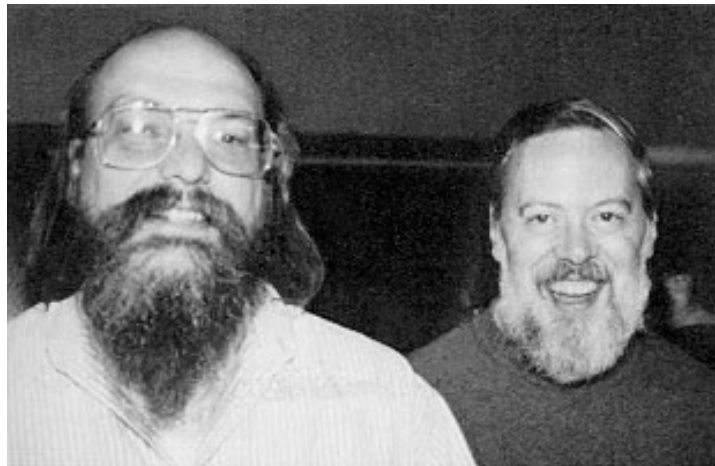


Figura 1: Ken Thompson i Dennis Ritchie

Dennis MacAlistair Ritchie va ser un informàtic estatunidenc, investigador als Bell Labs, conegut sobretot per dues aportacions fonamentals a la informàtica: el desenvolupament d'Unix (juntament amb Ken Thompson) i la creació del llenguatge de programació **C** (1972), que va utilitzar precisament per reescriure el nucli d'Unix i fer-lo portable a diferent maquinari.

El llenguatge C es va convertir en un dels llenguatges més influents de la història, base de molts altres llenguatges posteriors (C++, Java, C#, entre d'altres) i encara avui fonamental en el desenvolupament de sistemes operatius, controladors i programari de baix nivell. Per aquesta doble aportació (Unix i C), Ritchie va rebre el 1983 el **Premi Turing**, juntament amb Thompson, i la **National Medal of Technology** el 1999.

3.2 Richard Stallman (1953-)



Figura 2: Richard Stallman

Richard Matthew Stallman, sovint conegut per les inicials **RMS**, és un programador i activista estatunidenc, fundador del **projecte GNU** (1983) i de la **Free Software Foundation** (1985). És l'autor de l'anomenat "manifest GNU" i el principal redactor de la **Llicència GPL**.

Stallman va treballar al laboratori d'intel·ligència artificial del MIT, on va viure de primera mà la transició d'una cultura de programari compartit entre investigadors cap a un model cada cop més restrictiu i propietari. Aquesta experiència el va portar a formular les seves idees sobre el **programari lliure**, que distingeix explícitament de l'expressió "codi obert" (*open source*): mentre que el programari de codi obert se centra en avantatges pràctics (qualitat, seguretat, cost), per a Stallman el programari lliure és, sobretot, una qüestió ètica relacionada amb la llibertat de l'usuari.

A més del projecte GNU, és autor d'eines com l'editor **Emacs** i el compilador **GCC**, dues peces clau de l'ecosistema de programari lliure.

3.3 Linus Torvalds (1969-)



Figura 3: Linus Torvalds

Linus Benedict Torvalds és un enginyer de programari finlandès, creador del nucli **Linux**. El 1991, sent encara estudiant a la Universitat de Hèlsinki, va començar a desenvolupar aquest nucli com a projecte personal, inspirant-se en **Minix**, un sistema operatiu acadèmic de tipus Unix.

Torvalds va publicar el codi font del nucli i en va demanar opinió a la comunitat a través d'un grup de notícies (Usenet), un gest que va resultar decisiu: ràpidament es va formar al seu voltant una comunitat de desenvolupadors voluntaris que hi van anar contribuint. El 1992 va llicenciar Linux sota **GPL**, decisió que en va permetre la combinació amb les eines del projecte GNU, donant lloc al sistema **GNU/Linux**.

Torvalds continua avui sent el principal coordinador del desenvolupament del nucli Linux, i també és l'autor del sistema de control de versions **Git** (2005), creat inicialment per gestionar el mateix desenvolupament del nucli després d'un conflicte amb l'eina que s'utilitzava fins aleshores.

4. La llicència GPL

La **GPL** (*GNU General Public License*) és la llicència de programari lliure creada per Richard Stallman per protegir legalment les quatre llibertats del projecte GNU. La seva primera versió data de 1989, i la versió més utilitzada avui dia és la **GPLv2** (1991, sota la qual es distribueix el nucli Linux) i la **GPLv3** (2007, que actualitza la llicència davant de noves problemàtiques com les patents de programari o la *tivoització*).

La característica que distingeix la GPL d'altres llicències lliures és la clàusula de **copyleft**: qualsevol obra derivada d'un programa sota GPL ha de distribuir-se també sota GPL, amb el codi font disponible. Això garanteix que les llibertats originals es mantinguin en totes les versions futures del programari i no es puguin "tancar" en una versió propietària. Aquest mecanisme s'ha descrit sovint com un "judo legal": utilitza el mateix sistema de drets d'autor (*copyright*) per garantir precisament el contrari del que aquest sistema sol protegir.

Els punts clau de la GPL es poden resumir en:

Aspecte	Descripció
Llibertat d'ús	Es pot executar el programa amb qualsevol finalitat
Llibertat d'estudi	S'ha de poder accedir al codi font
Llibertat de redistribució	Es poden fer i compartir còpies
Llibertat de modificació	Es pot modificar el codi i distribuir les modificacions
Copyleft	Les obres derivades han de mantenir la mateixa llicència GPL

5. Característiques de GNU/Linux

A partir d'aquesta herència, GNU/Linux presenta un conjunt de característiques diferencials respecte a altres sistemes operatius:

5.1 Programari lliure i de codi obert

El nucli Linux i la majoria d'eines del sistema es distribueixen sota llicències lliures (principalment GPL), que garanteixen a l'usuari les quatre llibertats bàsiques: ús, estudi, modificació i redistribució. El codi font és públic i qualsevol persona pot inspeccionar-lo, auditar-lo o contribuir-hi.

5.2 Multiusuari i multitasca real

GNU/Linux és, des del seu disseny, un sistema **multiusuari** (diverses persones poden treballar-hi simultàniament, cadascuna amb el seu compte i permisos) i **multitasca preventiva** (el nucli pot interrompre l'execució d'un procés per donar temps de CPU a un altre, garantint que cap programa monopolitzi el sistema).

5.3 Multiplataforma

El nucli Linux funciona en una varietat de maquinari enorme: des de superordinadors i servidors x86_64, fins a telèfons mòbils (Android), routers, electrodomèstics, vehicles, satèl·lits o petits dispositius encastats (Raspberry Pi i similars).

5.4 Model de permisos i seguretat

GNU/Linux implementa un model de permisos robust basat en usuaris, grups i permisos de lectura/escriptura/execució (rwx) per a cada fitxer i directori, ampliable amb [ACL](#) (l·listes de control d'accés). L'usuari root té privilegis administratius totals, mentre que la resta d'usuaris operen amb permisos restringits, fet que limita l'impacte d'errors o programari maliciós.

5.5 Gran varietat de distribucions

A diferència d'altres sistemes operatius, GNU/Linux no és un producte únic, sinó un **nucli** sobre el qual es construeixen multitud de **distribucions** (Debian, Fedora, Ubuntu, Arch Linux, openSUSE, etc.), cadascuna amb les seves pròpies eines de gestió de paquets, filosofia i públic objectiu.

5.6 Gestió de paquets centralitzada

La majoria de distribucions ofereixen un sistema de **gestió de paquets** (com apt a Debian/Ubuntu o dnf a Fedora) que permet instal·lar, actualitzar i eliminar programari de manera centralitzada, resolent automàticament les dependències entre paquets.

5.7 Estabilitat i rendiment

Gràcies al seu disseny i a dècades d'optimització, GNU/Linux és conegut per la seva estabilitat (és habitual trobar servidors amb mesos o anys sense reiniciar-se) i el seu bon rendiment, especialment en entorns de servidor, cosa que explica el seu domini absolut en aquest àmbit i també en supercomputació.

5.8 Personalització i flexibilitat

Pràcticament tots els components del sistema són configurables o substituïbles: des de l'entorn d'escriptori (GNOME, KDE, XFCE...) fins al mateix nucli, que es pot recompilar amb opcions específiques. Això fa de GNU/Linux un sistema molt adaptable, des de servidors minimalistes sense interfície gràfica fins a estacions de treball completes.

5.9 Línia d'ordres potent

Tot i disposar d'entorns gràfics moderns i complets, GNU/Linux manté la línia d'ordres (shell) com a eina central per a l'administració del sistema, amb intèrprets com **Bash** que permeten automatitzar tasques mitjançant scripts, seguint l'esperit de la filosofia Unix descrita a l'apartat anterior.

6. Resum

Unix va néixer el 1969 com una alternativa senzilla a Multics i va consolidar, amb el pas dels anys, una filosofia de disseny basada en la simplicitat, la modularitat i la combinació d'eines petites i especialitzades. El projecte GNU, iniciat per Richard Stallman el 1983, va voler recrear aquest model amb programari íntegrament lliure, i la peça que li faltava ---un nucli--- la va aportar Linus Torvalds el 1991 amb Linux. D'aquesta confluència sorgeix GNU/Linux, un sistema operatiu lliure, multiusuari, multiplataforma i extremadament flexible, que avui dia és la base de la majoria de servidors i sistemes encastats del món.

7. Referències

Sobre la història d'Unix

- The Unix Heritage Society (TUHS) --- arxiu històric de codi font i documentació de totes les versions d'Unix: <https://www.tuhs.org/>
- Dennis M. Ritchie, *The Evolution of the Unix Time-sharing System* (paper original sobre l'evolució del sistema): https://amturing.acm.org/bib/ritchie_1506389.cfm

Sobre la filosofia Unix

- Eric S. Raymond, *The Art of Unix Programming* (text complet disponible en línia): <http://www.catb.org/esr/writings/taoup/html/>

Sobre GNU i la GPL

- Projecte GNU, lloc oficial: <https://www.gnu.org/>
- *The GNU Manifesto*, Richard Stallman (1985): <https://www.gnu.org/gnu/manifesto.html>
- Text complet de la GPL v3: <https://www.gnu.org/licenses/gpl-3.0.en.html>
- Free Software Foundation, lloc oficial: <https://www.fsf.org/about>

Sobre el kernel Linux

- kernel.org --- lloc oficial i punt de distribució del nucli Linux: <https://www.kernel.org/>

Sobre les distribucions de GNU/Linux

- DistroWatch --- directori i notícies de centenars de distribucions GNU/Linux i BSD: <https://distrowatch.com/>
- Arbre genealògic visual de distribucions (Visual Distro Family Tree) a DistroWatch: <https://distrowatch.com/dwres.php?resource=family-tree>

Sobre els protagonistes

- ACM A.M. Turing Award --- fitxa de Dennis M. Ritchie (premi compartit amb Ken Thompson, 1983): https://amturing.acm.org/award_winners/ritchie_1506389.cfm
- Lloc personal de Richard Stallman: <https://stallman.org/>

Versions d'aquest document

- HTML - [historia.html](#)
- PDF - [historia.pdf](#)
- ODT - [historia.odt](#)
- MD - [historia.md](#)

[Domini Públic \(CC0\)](#)