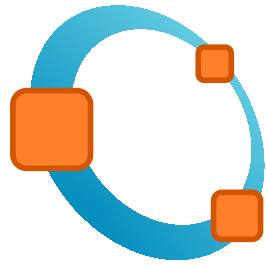




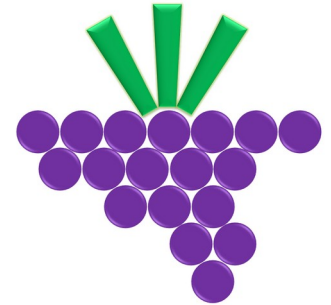
Introducción a Octave



para ciencias aplicadas e ingeniería



Unidad 0



San Rafael, Argentina



UNCUYO
UNIVERSIDAD
NACIONAL DE CUYO



FACULTAD DE
**CIENCIAS APLICADAS
A LA INDUSTRIA**

CONICET

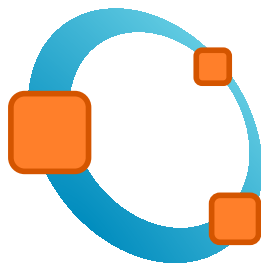


MOCCAI
MODELADO COMPUTACIONAL EN CIENCIAS APLICADAS E INGENIERÍA



Introducción a Octave

para ciencias aplicadas e ingeniería



Daniel Millán

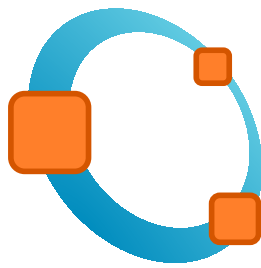


Nicolas Muzi



Introducción a Octave

para ciencias aplicadas e ingeniería



- Curva sigmoide, arrancamos lento y luego pega una subida importante
- Entrega obligatoria de problemas seleccionados
- Asistencia 80% de las clases de teoría
- Asistencia 80% de las clases de práctica (consultas opcionales)
- La idea es desarrollar el trabajo en clase
- Última de guía de problemas de mayor complejidad y evaluación más puntillosa





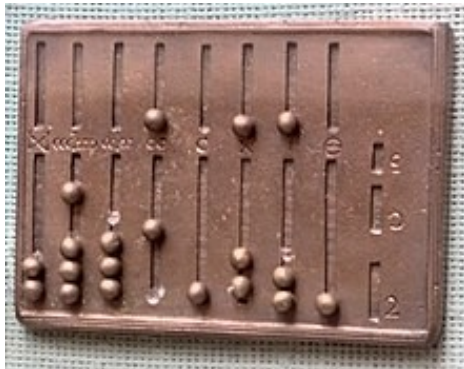
Introducción

En la antigua Mesopotamia, en la civilización Sumeria, tuvo su origen el **sistema sexagesimal**, es un sistema de numeración posicional que emplea como base aritmética el número **60**.

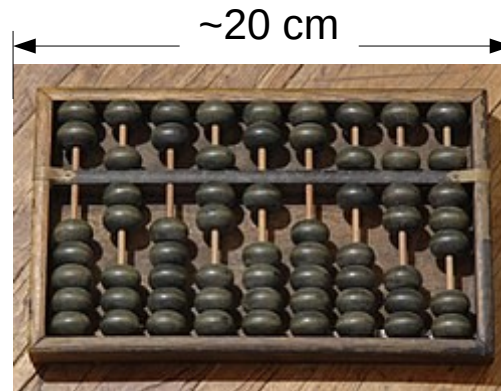
El sistema sexagesimal se usa para medir tiempos (horas, minutos y segundos) y ángulos (grados) principalmente.

Ábaco: permite realizar operaciones aritméticas sencillas, su origen se remonta a la antigua Mesopotamia, más de 2000 años antes de nuestra era.

Wikipedia



≤200 a.C.: Ábaco Romano.



≤200 a.C.: Suanpan es un ábaco de origen chino.





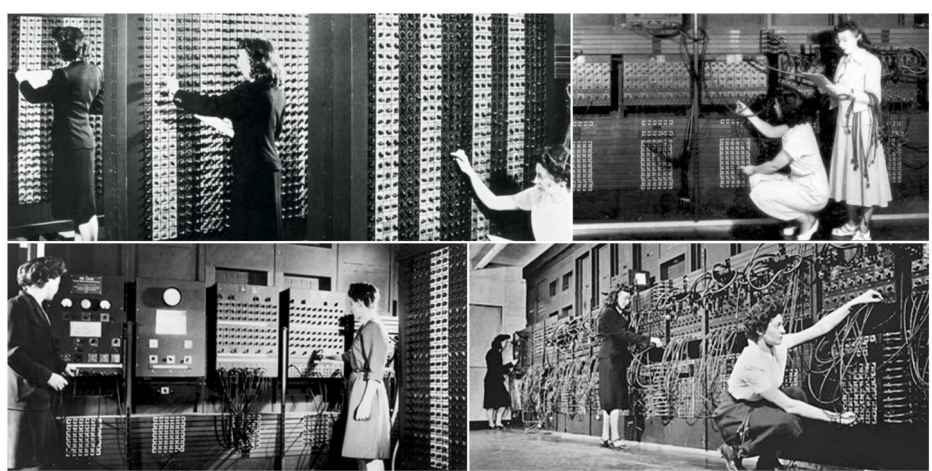
Introducción



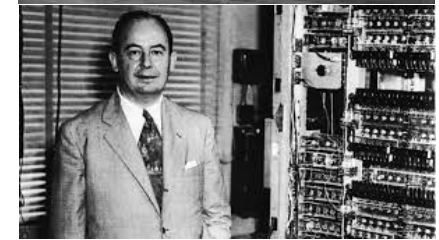
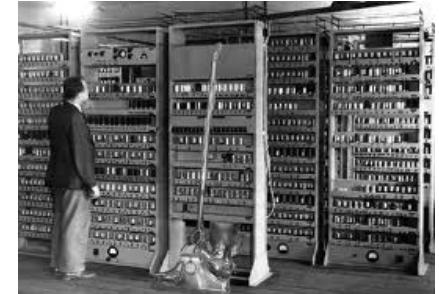
1645: Blaise Pascal inventa la **pascalina**, una de las primeras calculadoras mecánicas. Funcionaba a base de ruedas de diez dientes, cada uno representaba un dígito del 0 al 9.

1949: La **EDVAC** fue la primer computadora de programas almacenados electrónicamente en forma binaria.

Wikipedia.



1946: La **ENIAC** fue inicialmente diseñada para calcular tablas de tiro de artillería destinadas al Laboratorio de Investigación Balística de la Universidad de Pensilvania, para el ARMY USA.



John von Neumann



2010: Scan 3XS Cyclone PC

primer NVIDIA GeForce GTX 480 con refrigeración líquida

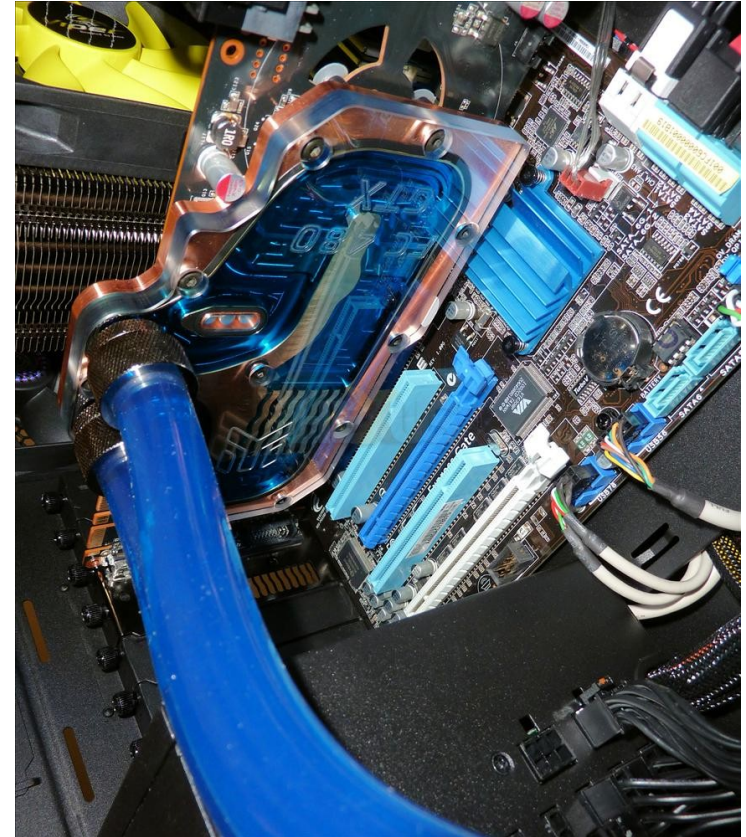


<http://hexus.net/tech/reviews/systems/25017-scan-3xs-cyclone-pc-first-watercooled-geforce-gtx-480/>



2010: Scan 3XS Cyclone PC

- primer tarjeta gráfica de NVIDIA con refrigeración líquida
- overclocked GeForce GTX 480, opera a 852MHz (701MHz)
- procesador i7 920, *overclocked* a 4GHz
- £1,646.84, incluyendo impuestos





2020: MareNostrum Barcelona



<https://www.bsc.es/>

**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

- *MareNostrum* es el supercomputador más potente de España, el 5º más rápido de Europa y el 29º del mundo, según la lista **TOP500** (nov - 2018).





2020: MareNostrum Barcelona



Áreas de investigación

- **Composición atmosférica:** calidad del aire, aerosoles y como estos dispersan y absorben la radiación solar, ciudades inteligentes y la optimización del transporte y la salud humana.
- **Big Data:** herramientas visuales y algorítmicas para analizar y estudiar grandes volúmenes de datos.
- **Bioinformática:** integración, almacenamiento y transmisión de gran volumen de datos clínicos y datos de simulaciones, diseño de fármacos.
- **Biomecánica:** sistema cardiovascular y sistema respiratorio.
- **Predicción climática:** gestión de la agricultura y del agua, el pronóstico oceánico, estudio de los ciclones tropicales, estudio de dónde es más eficiente instalar un molino de viento.
- **Computación en la nube:** informática energética y optimización de los centros de datos.



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación

<https://www.bsc.es/>

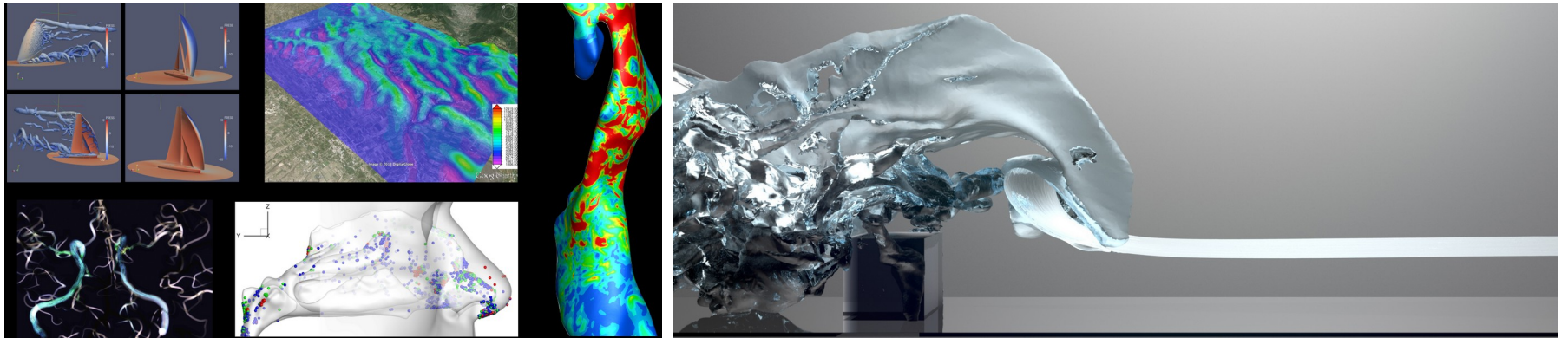


2020: MareNostrum Barcelona



Áreas de investigación

- **Simulación de ingeniería:** reducción de las emisiones contaminantes, computación en mecánica de fluidos, mecánica no lineal de sólidos.



- **Geofísica:** terremotos, detección de la presencia de fluidos a grandes profundidades bajo la superficie de la Tierra, propiedades de la superficie de la Tierra.

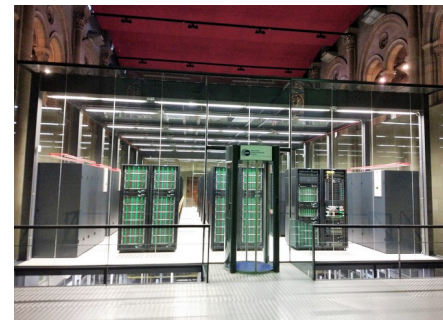
Simulación social: evolución cultural, eficiencia energética, seguridad pública de cara a tener ciudades inteligentes y resistentes.





2020: MareNostrum Barcelona

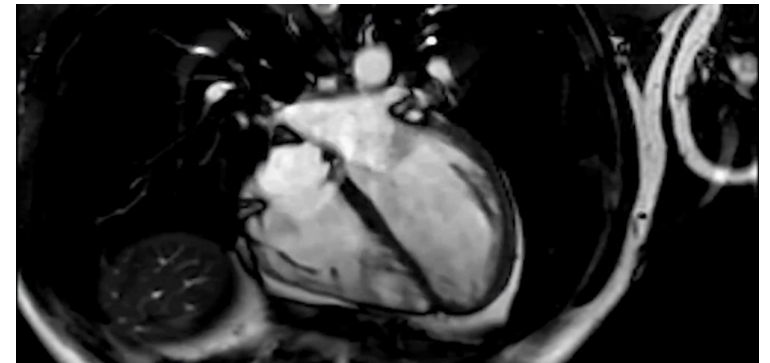
Alya Red



Proyecto **Alya Red**, su video promocional fue **elegido mejor vídeo científico del 2012** por la *National Science Foundation* de EEUU y la revista *Science*. En él se explica cómo se crean los modelos con los que se simula el funcionamiento de un corazón, intentando imitar el comportamiento de los diferentes tejidos y de cómo las señales eléctricas viajan por su interior.

La tarea es tan compleja que para poder analizarlo con precisión se emplea el ordenador Mare Nostrum del Centro de Supercomputación de Barcelona (BSC).

<https://www.youtube.com/watch?v=gaG21uZrZ3E>

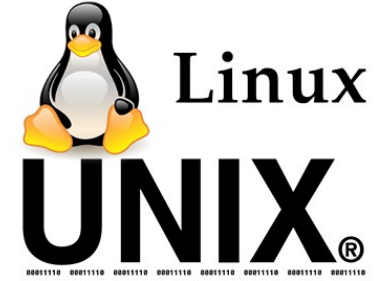




Unidad 0

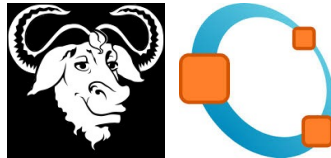
1. ¿Qué es un sistema operativo?

2. Breve historia de Unix



3. Arquitectura del SO Linux

4. GÑU Octave





1. ¿Qué es un OS?





1. ¿Qué es un OS?

- Un **sistema operativo (OS)** es un gestor (administrador) de recursos
- Se presenta en forma de un conjunto de rutinas de software que permiten a los usuarios y a los programas acceder a los recursos del sistema de una manera **segura, eficiente y abstracta**
 - CPU, tarjetas de red, discos de memoria, módems, impresoras, etc...
 - **CPU**: *central processing unit*
Unidad de Procesamiento Central
 - El OS asegura un acceso seguro p.ej. impresora
 - El OS fomenta el uso eficiente de la CPU mediante suspensión de operaciones de *Entrada/Salida*
 - El OS proporciona abstracciones tales como archivos en lugar de posiciones de memoria en discos (detalles de hardware están ocultos)



Wikipedia





1. ¿Qué es un OS?

- El **núcleo del OS** controla de forma directa el hardware subyacente
- El núcleo maneja dispositivos de bajo nivel, la memoria y la gestión del procesador
- Servicios básicos del núcleo están disponibles para programas de nivel superior a través de una biblioteca de **llamadas al sistema**
- **Los programas informáticos o aplicaciones** (procesadores de texto, hojas de cálculo, **Octave**) y **programas de utilidades del sistema** (buscador) hacen uso de las llamadas al sistema
- Aplicaciones y utilidades del sistema se ponen en marcha mediante un **shell** (interfaz de órdenes de texto) o una **interfaz gráfica de usuario** que proporciona una interacción directa (mouse)





2. Breve historia de Unix

Principios de diseño

- UNIX fue diseñado para ser un **SO interactivo, multiusuario y multitarea**:
 - **Interactivo** quiere decir que el sistema acepta órdenes, las ejecuta y se dispone a esperar otras nuevas.
 - **Multitarea** significa que puede realizar varios trabajos, denominados procesos, al mismo tiempo.
 - **Multiusuario** significa que más de una persona puede usar el sistema al mismo tiempo.
- UNIX fue diseñado por programadores para ser usado por programadores en un entorno en que los usuarios son relativamente expertos y participan en el desarrollo de proyectos de software




UNIX → No user friendly



2. Breve historia de Unix

American Telephone and Telegraph
(Direct TV U\$S 170G #9 USA Fortune 500)

- **1960:** General Electric + MIT + Bell Labs (**AT&T**) desarrollan MULTICS
 - SO multi-usuario y multitarea en ordenadores centrales (cajas grandes)
 - MULTICS: ***MULT**iplexed **I**nformation and **C**omputing **S**ystem*
- **1969:** **Ken Thompson** (Bell Labs)
 - Crea un SO basado en MULTICS pero más sencillo en una **PDP-7** (mini PC 1965)
 - UNICS: ***UN**iplexed **I**nformation and **C**omputing **S**ystem*
 - Poca memoria y potencia llevan a utilizar comandos cortos: **ls**, **cp**, **mv**...
 - El lenguaje de programación en que fue escrito UNICS se llamaba B
- **1971:** Se une **Dennis Ritchie**
 - Crea el primer compilador de C y se reescribe el núcleo de **UNIX en C** (1973)
 - Mejora de la portabilidad
 - Se lanza la quinta versión de UNIX a las Universidades en 1974 (GRATIS)
- **1978:** Se separan dos grandes ramas: SYSV (AT&T y otras empresas) y BSD (Berkeley Software Distribution de la UCB) → Incompatibles!

In 1975, [Ken Thompson](#) took a [sabbatical](#) from Bell Labs and came to Berkeley as a visiting professor, he helped to install Version 6 Unix .





2. Breve historia de Unix

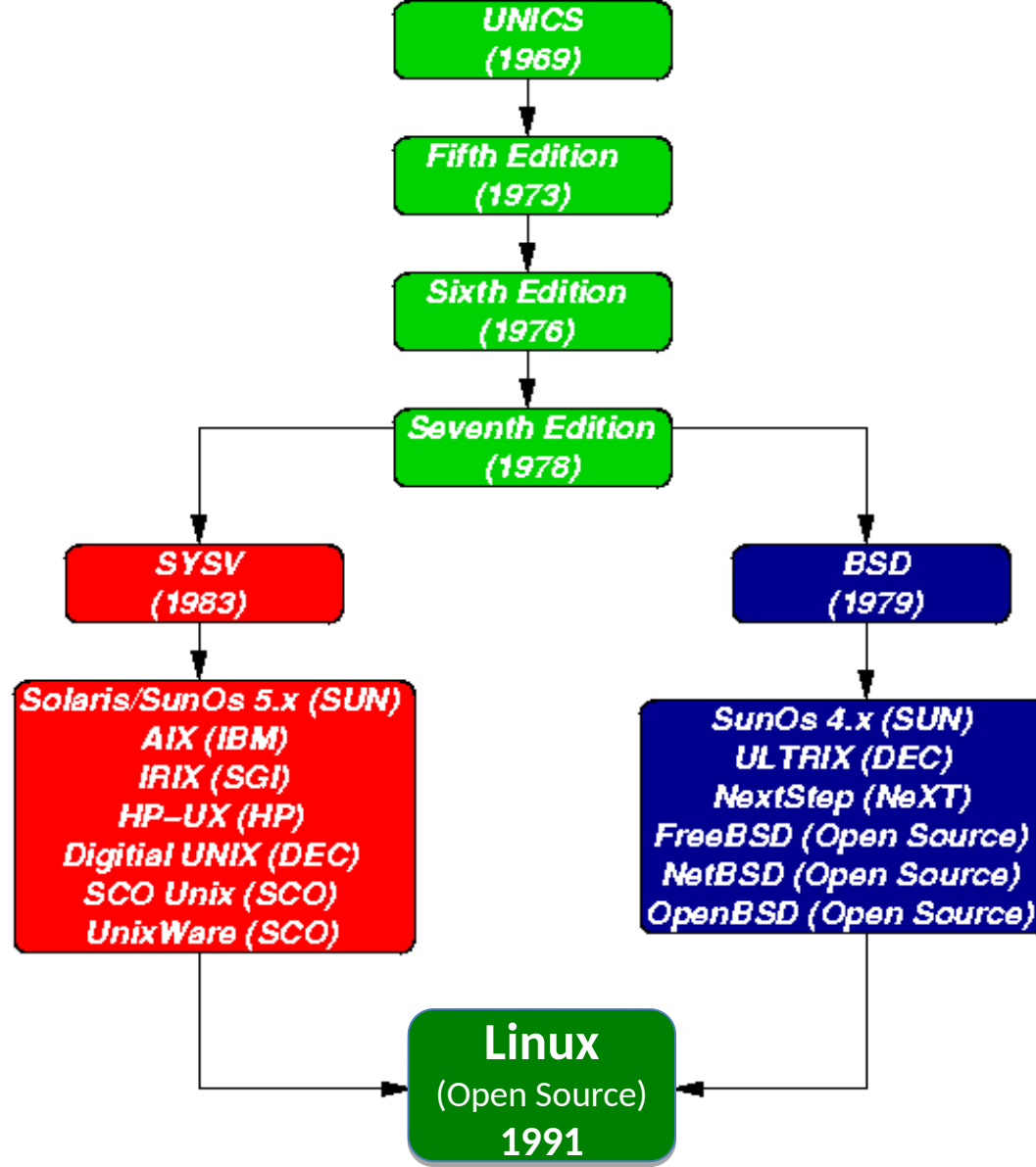


Un **PDP-7** modificado, en restauración en Oslo, Noruega. [Wikipedia](#)



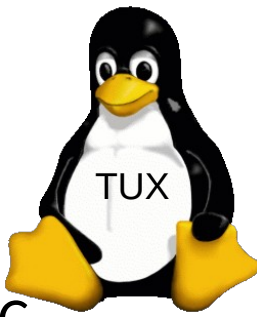
Ken Thompson y Dennis Ritchie. [Wikipedia](#)







2. Breve historia de Unix



- **1991:** Linus Torvalds, 21 años, estudiante finlandés de Ciencias de la Computación diseña Linux un código abierto del SO UNIX para PC

- No es SYSV ni BSD, pero incorpora características de cada uno (p.ej. al estilo SYSV archivos de inicio, pero con una disposición del sistema de archivos del tipo BSD)
- Cumple con un conjunto de estándares de IEEE (*Institute of Electrical and Electronics Engineers*) llamado POSIX (*Portable Operating System Interface*)
- Para maximizar la portabilidad del código, Linux *típicamente* soporta SYSV, BSD y llamadas al sistema de POSIX
- Linux ha generado que miles de personas colaboren voluntariamente durante **30 años** mejorando el núcleo y programas de aplicación
- Diferentes distribuciones: Debian, Suse, RedHat, Ubuntu, etc
- Portable a diferentes arquitecturas de procesadores como Intel, AMD, SPARC...
- *Fácil* de usar e instalar y viene con un conjunto completo de utilidades y aplicaciones, incluyendo el sistema de gráficos X, entornos GNOME y KDE GUI

In **1996** Larry Ewing released Tux under the following condition: **TUX:** (T)orvalds (U)ni(X) or tuxedo abbreviation... how knows?
Permission to use and/or modify this image is granted provided you acknowledge me lewing@isc.tamu.edu and The GIMP if someone asks.



17 de septiembre

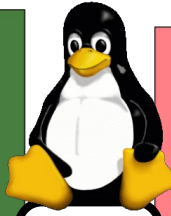




Unnamed PDP-7 OS

■ Open Source
■ Mixed/Shared Source
■ Closed Source

1969
 1971 to 1973
 1974 to 1975
 1978
 1979
 1980
 1981
 1982
 1983
 1984
 1985
 1986
1987
 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 2013



Linux

NEXTSTEP/
 OPENSTEP
 1.0 to 4.0

Mac OS X Server

Mac OS X
 0.0 to 10.9.x
 (Darwin)

FreeBSD 1.0 to 2.2.x

FreeBSD 3.0 to 3.2

FreeBSD 3.3 to 9.x

BSD 4.4 to 4.4 lite2

NetBSD 1.3 to 6.x

OpenBSD 1.0 to 2.2

OpenBSD 2.3 to 5.x

Sun OS 1 to 1.1

Sun OS 1.2 to 3.0

Sun OS 4

AIX 1.0

AIX 3.x to 7.1

SCO Unix 3.2.4

OpenServer 5.0 to 5.04

OpenServer 5.0.5 to 5.0.7

OpenServer 6.0

UnixWare 1.x to 2.x

UnixWare 7.x

Solaris 2.1 to 9

Solaris 10

Solaris 11

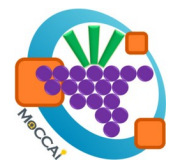
OpenSolaris and derivatives

HP-UX 1.0 to 1.2

HP-UX 2.0 to 3.0

HP-UX 6 to 11

HP-UX 11i to 11i v3





3. Arquitectura del SO Linux

Linux tiene todos los componentes de un SO tipo UNIX:

- Núcleo: facilita acceso seguro a distintos programas al *hardware* (tarjetas gráficas y red, discos duros, etc), decide qué programas utilizan hardware y cuánto tiempo (multiplexado), BSD/SYSV llamadas de sistema, etc
- Shells y GUIs:
 - Intérpretes de línea de comandos (shells) como en UNIX:
 - **sh**: shell Bourne, **bash**: *Bourne again shell* y **cs****h**: *C shell*
 - Interface Gráfica (GUI, *Graphic User Interface*), gestores KDE y GNOME
- Utilidades del sistema: Herramientas poderosas que hacen una sola tarea extremadamente bien.
 - **cp** copia
 - **grep**: busca expresiones regulares (caracteres)
 - **awk**: procesa datos definidos en archivos de texto
 - **sed**: editor de flujo de texto, demonios, etc

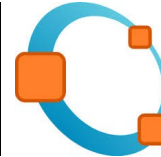




3. Arquitectura del SO Linux

Linux tiene todos los componentes de un SO tipo UNIX:

- Programas de aplicación:
 - **emacs**: editor de texto
 - **gcc/g++**: compilador de C/C++
 - **latex**: lenguaje de composición de texto
 - **GÑU Octave**: cálculos matriciales/vectoriales





<https://introoctave.github.io/>

- **Lecturas para curiosos (wiki++)**
- **CAPÍTULO IV – UN SISTEMA DEL QUE DERIVARLOS A TODOS**
<https://www.ionlitio.com/hackers-capitulo-iv/>

- **CAPÍTULO V – UN PINGÜINO LLAMADO TUX**
<https://www.ionlitio.com/hackers-capitulo-v/>

- **The Art of Unix Programming**
<http://www.faqs.org/docs/artu/index.html>

All the philosophy really boils down to one iron law, the hallowed ‘KISS principle’ of master engineers everywhere:



K.I.S.S.

Keep It Simple, Stupid!

