Historia del Software

De la lógica mecánica a la inteligencia artificial generativa

Una Masterclass sobre la evolución del sistema operativo del mundo.

AGENDA: EL VIAJE EN 10 ERAS

- Los Cimientos (1843-1940)
- La Era del Código Máquina (1940s)
- Nacimiento del Alto Nivel (1950s)
- La Crisis de Software (1960s)
- La Revolución Estructurada (1970s)
- El PC y la GUI (1980s)
- La Explosión de la Web (1990s)
- La Era Ágil y el Mobile (2000s)
- La Nube y el Big Data (2010s)
- El Futuro de la IA (2020s+)

Sección I

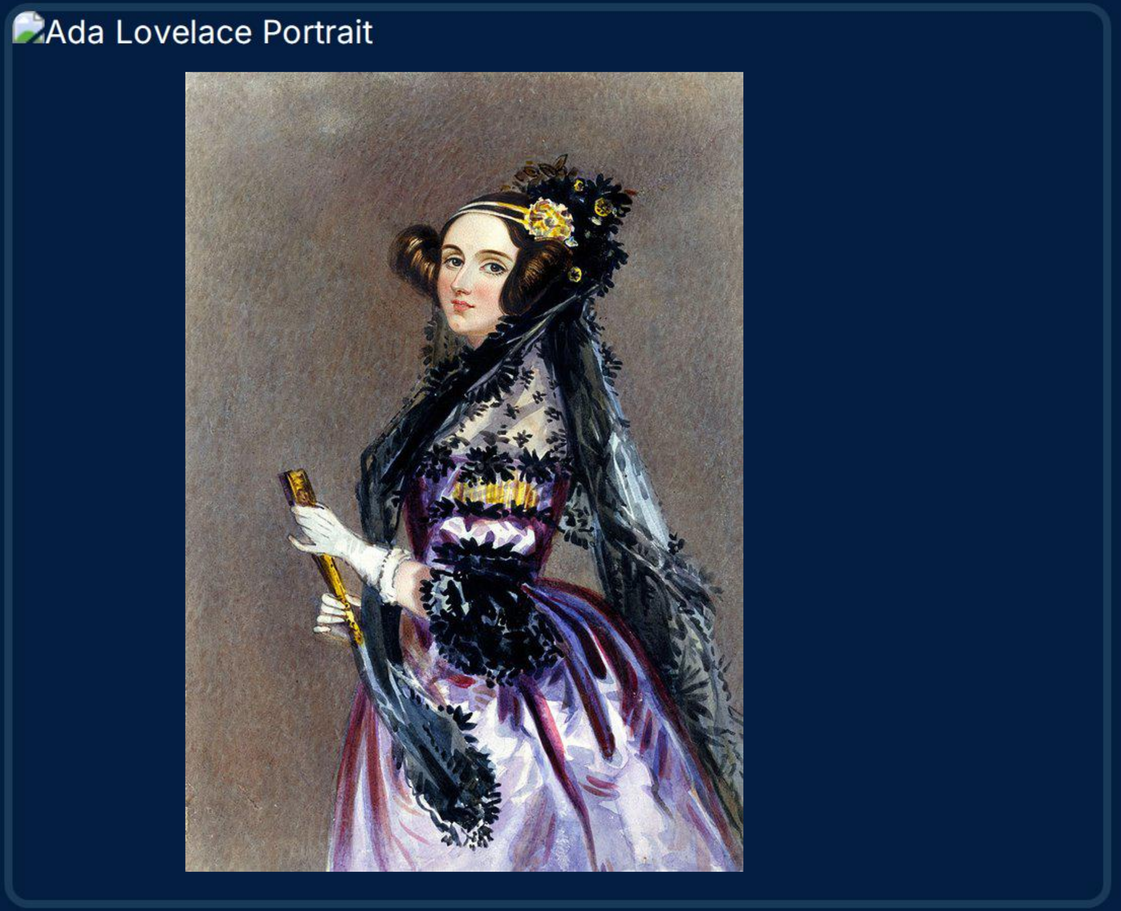
Los Cimientos (1843 - 1940)

Cuando el software era solo una idea matemática.

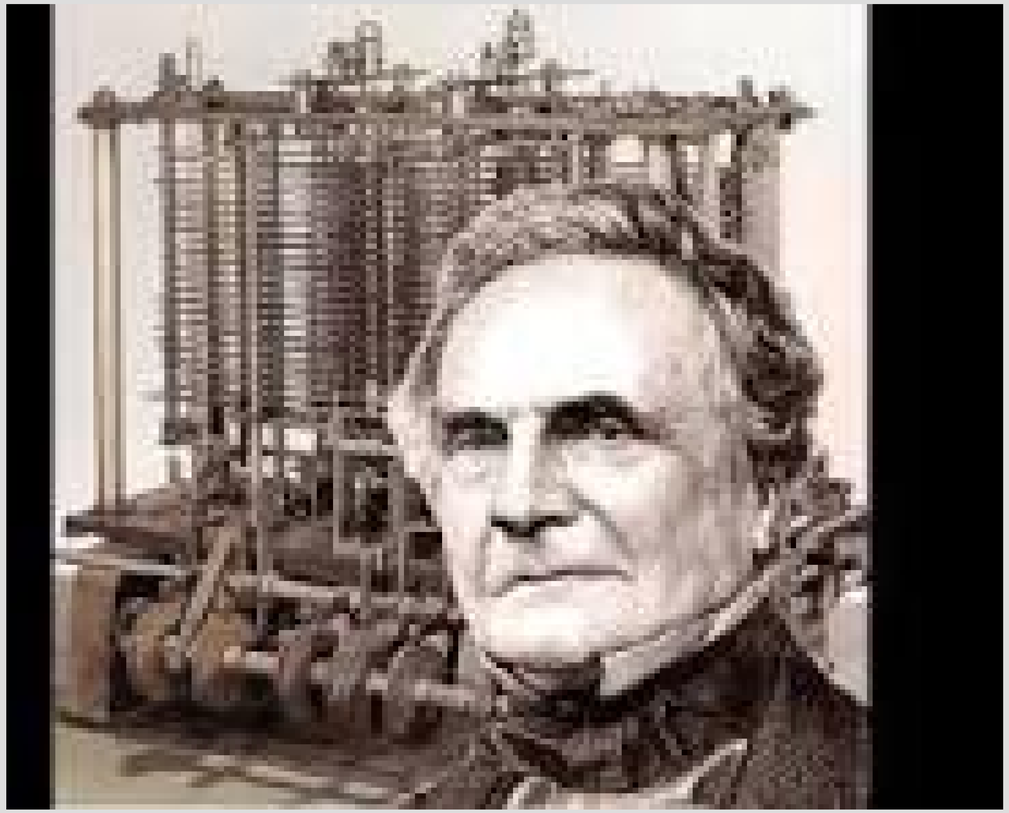
ADA LOVELACE: LA PRIMERA PROGRAMADORA

En **1843**, Ada Lovelace tradujo las notas de Luigi Menabrea sobre la Máquina Analítica de Babbage.

- Escribió el primer algoritmo diseñado para ser procesado por una máquina.
- Visualizó la capacidad de las máquinas para procesar más que números (música, gráficos).
- Sentó las bases de la **computabilidad**.



CHARLES BABBAGE: EL HARDWARE QUE ESPERABA



El diseño de la **Máquina Analítica** incluía elementos modernos del software:

- Entrada por tarjetas perforadas (datos).
- Memoria para almacenar resultados.
- Unidad central de procesamiento (MOLINO).
- Control de flujo mediante bucles y saltos condicionales.

ALAN TURING: LA MÁQUINA UNIVERSAL

En **1936**, Turing publicó "Sobre números computables", introduciendo la Máquina de Turing.

> *"Cualquier proceso algorítmico puede ser realizado por una máquina universal si este es computable."*

Turing definió que el software es independiente del hardware; una idea revolucionaria.

Sección II

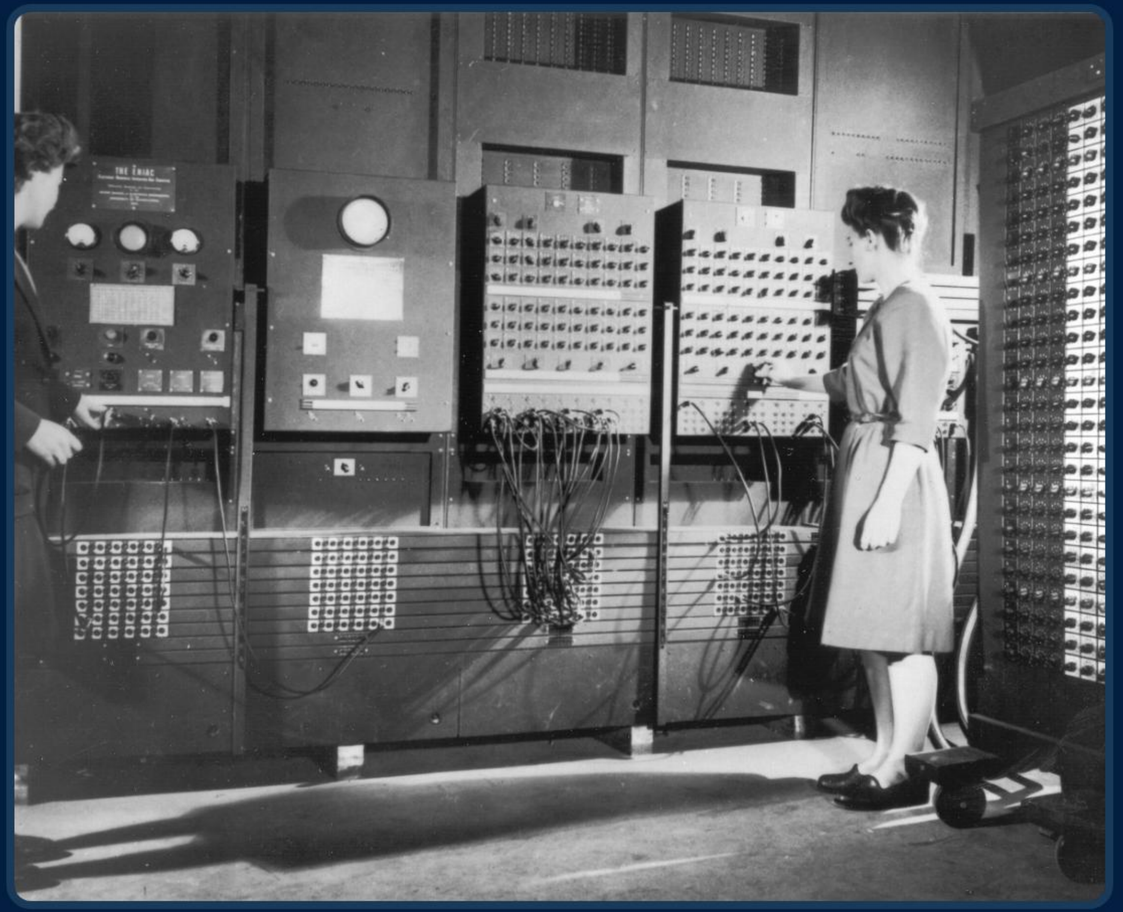
La Era del Código Máquina (1940 - 1950)

Del cableado físico al programa almacenado.

ENIAC: PROGRAMACIÓN POR CABLEADO

En **1946**, el ENIAC era una mole de 30 toneladas.

- No había teclado ni pantalla.
- La programación consistía en conectar cables en tableros.
- Un cambio en el "software" podía tomar días de labor física.



LAS PROGRAMADORAS INVISIBLES

Seis mujeres (McNulty, Jennings, Snyder, Wescoff, Meltzer y Bilas) fueron las responsables de que el ENIAC funcionara.

Lógica Matemática	Depuración Manual	Estructura de Datos

Fueron pioneras en el concepto de **depuración de software**.

ARQUITECTURA VON NEUMANN (1945)

John von Neumann propuso el **Programa Almacenado** en el reporte EDVAC.

- Los datos y las instrucciones se almacenan en la misma memoria.
- Elimina la necesidad de recablear físicamente la máquina.
- Nace el concepto moderno de **software cargable**.

LENGUAJE ENSAMBLADOR: LA PRIMERA CAPA

Programar en binario (0101) era insostenible.

```
MOV EAX, 1
ADD EAX, 2
INT 0x80
```

- Uso de **mnemónicos** para representar instrucciones.
- Sigue ligado 1:1 al hardware.
- Permitió aumentar la complejidad de los programas.

Sección III

El Nacimiento del Alto Nivel (1950 - 1960)

Cuando las palabras humanas empezaron a controlar máquinas.

GRACE HOPPER Y EL COMPILADOR

En **1952**, Grace Hopper creó el sistema A-0.

- Primer programa que traduce lenguaje simbólico a código máquina.
- Lideró el desarrollo de **COBOL** (lenguaje orientado a negocios).
- Acuñó el término "bug" tras encontrar una polilla en un relé.

JOHN BACKUS Y FORTRAN (1957)

El **FORmula TRANslation** fue el primer lenguaje de programación de alto nivel exitoso.

- Diseñado para científicos e ingenieros (cálculos numéricos).
- Redujo el número de instrucciones necesarias en un 2000%.
- Hizo que el software fuera independiente de la CPU específica por primera vez.

LISP Y LA INTELIGENCIA ARTIFICIAL (1958)

John McCarthy creó **LISP** en el MIT.

- Introdujo el manejo de listas y la recursión.
- Se convirtió en el lenguaje estándar para la IA durante décadas.
- Pionero en la **recolección de basura** (garbage collection).

ALGOL: EL PADRE DE LA ESTRUCTURA

ALGOL 58/60 fue la raíz de casi todos los lenguajes modernos (Pascal, C, Java).

- Introdujo bloques de código (BEGIN/END).
- Definió la **notación BNF** para gramáticas de lenguajes.
- Primer lenguaje portado a múltiples arquitecturas.

Sección IV

La Crisis del Software y la Ingeniería (1960 - 1970)

El caos de la complejidad obliga a crear una ciencia.

LA CRISIS DEL SOFTWARE (1968)

En la década de los 60, el hardware superó al software.

- Costos Incontrolables
- Retrasos Masivos
- Código Espagueti

El software se volvió demasiado complejo para ser manejado por "artesanos".

NACE LA "INGENIERÍA DE SOFTWARE"

En la conferencia de la OTAN en **Garmisch, Alemania** (1968):

- Se propuso aplicar principios de ingeniería al desarrollo de programas.
- Se buscó la fiabilidad, eficiencia y mantenibilidad.
- Fue el paso de la "programación" a la "ingeniería".

IBM OS/360 Y EL MES MÍTICO

El desarrollo del OS/360 fue el proyecto de software más grande de su tiempo.

> *"Añadir mano de obra a un proyecto de software retrasado, lo retrasa aún más."*
> *— Fred Brooks*

Se aprendió que la comunicación en el equipo es el mayor cuello de botella.

MARGARET HAMILTON Y EL APOLLO 11

Lideró el equipo de software del MIT para el programa Apollo.

- Acuñó el término **Software Engineering** para ganar respeto frente al hardware.
- Diseñó el software de recuperación de errores que salvó el alunizaje del Apollo 11.

Sección V

La Revolución Estructurada (1970 - 1980)

Nacen UNIX, C y las bases de datos modernas.

RITCHIE, THOMPSON Y UNIX

En Bell Labs (**1971**), crearon el lenguaje C y el sistema operativo UNIX.

- **UNIX:** Portable, multiusuario y multitarea.
- **Lenguaje C:** El lenguaje de sistemas más influyente de la historia.

EDSGER DIJKSTRA Y LA ESTRUCTURA

Dijkstra publicó "Go To Statement Considered Harmful" en 1968.

- Promovió la **Programación Estructurada**.
- Uso de condicionales (if/then) y bucles (while/for) en lugar de saltos aleatorios.
- Buscó la verificación matemática de los algoritmos.

SMALLTALK: OBJETOS PUROS (1972)

En Xerox PARC, Alan Kay y su equipo crearon **Smalltalk**.

- Sentó las bases de la Programación Orientada a Objetos (POO).
- Todo es un objeto que envía y recibe mensajes.
- Influencia directa en Objective-C, Java y Python.

EDGAR F. CODD Y EL SQL (1970)

Codd propuso el **Modelo Relacional** de datos.

- Separó la forma en que se almacenan los datos de cómo se consultan.
- Dio origen a SQL (Structured Query Language).
- Permitió al software manejar volúmenes masivos de información de forma estructurada.

Sección VI

El Software Personal y la GUI (1980 - 1990)

El software llega a los hogares y oficinas.

MS-DOS Y EL AUGE DE MICROSOFT

En **1981**, Microsoft licenció MS-DOS para IBM.

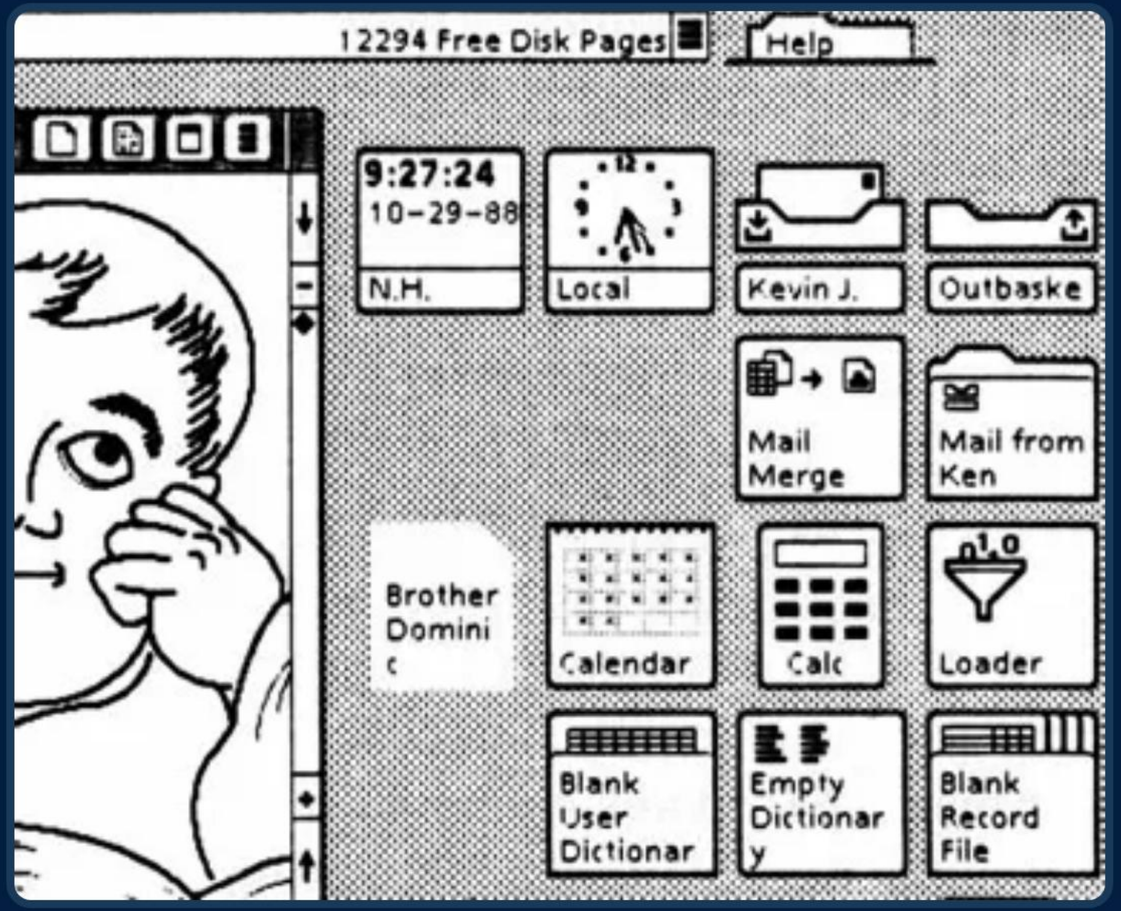
- Sistema de línea de comandos.
- Estandarizó el software para el PC compatible.
- Microsoft se convirtió en el gigante del software mundial.

```
C:\> DIR
C:\> CD DOS
C:\> FORMAT A: /S
```

XEROX PARC Y APPLE MACINTOSH

En **1984**, Apple lanzó el Macintosh.

- Popularizó la Interfaz Gráfica de Usuario (GUI).
- Uso del ratón, ventanas e iconos.
- El software dejó de requerir memorizar comandos.



BJARNE STROUSTRUP Y C++ (1983)

C++ añadió clases y orientación a objetos al lenguaje C.

- Permitió manejar la creciente complejidad de las interfaces gráficas.
- Mantuvo el rendimiento del lenguaje C.
- Lenguaje estándar para sistemas, videojuegos y software comercial por décadas.

RICHARD STALLMAN Y EL SOFTWARE LIBRE

En **1985**, nació la Free Software Foundation (FSF).

- Manifiesto GNU: Libertad para usar, estudiar, modificar y distribuir.
- Nace la licencia GPL.
- Un movimiento ético que cambió la industria para siempre.

Sección VII

La Explosión de la Web (1990 - 2000)

El software se vuelve hiperconectado.

BERNERS-LEE Y EL WORLD WIDE WEB

En **1990**, en el CERN, nació la Web.

- Primer servidor web (NeXT computer).
- Definición de HTML y HTTP.
- El software pasó de ser local a ser distribuido globalmente.

JAVA: PORTABILIDAD EXTREMA (1995)

Sun Microsystems lanzó Java con la premisa **"Write once, run anywhere"**.

- Introdujo la Máquina Virtual de Java (JVM).
- Manejo automático de memoria.
- Se convirtió en el estándar para aplicaciones empresariales y la web interactiva temprana.

LINUS TORVALDS Y LINUX (1991)

Un estudiante finlandés lanzó el kernel Linux como hobby.

- Se unió al proyecto GNU de Stallman.
- Demostró el poder del **desarrollo colaborativo** masivo.
- Hoy mueve casi todo internet y el 100% de las supercomputadoras.

GUIDO VAN ROSSUM Y PYTHON

Python (1991) priorizó la **legibilidad del código**.

- Sintaxis limpia y poderosa.
- Filosofía "Baterías incluidas".
- Décadas más tarde, se convertiría en el lenguaje #1 para Ciencia de Datos e IA.

JAVASCRIPT: 10 DÍAS QUE CAMBIARON LA WEB

Brendan Eich creó JavaScript en mayo de **1995** para Netscape.

- Añadió interactividad al navegador.
- A pesar de un inicio humilde, hoy es el lenguaje más usado en el mundo gracias a Node.js y React.

Sección VIII

La Era Ágil y el Mobile (2000 - 2010)

Nuevos procesos para un mundo rápido.

EL MANIFIESTO ÁGIL (2001)

17 desarrolladores se reunieron en Utah para buscar alternativas a los procesos pesados.

1. Individuos e interacciones sobre procesos y herramientas.
2. **Software funcional** sobre documentación extensiva.

Nacen Scrum, XP y el desarrollo iterativo.

SOFTWARE AS A SERVICE (SAAS)

El software dejó de comprarse en cajas.

- Salesforce (1999) lideró el cambio.
- Actualizaciones constantes en lugar de versiones anuales.
- El navegador se convirtió en el **entorno de ejecución** principal.

IPHONE Y ANDROID: SOFTWARE TÁCTIL

En **2007/2008**, nació el software móvil masivo.

- Apps nativas y App Stores.
- Nuevo paradigma de interfaz de usuario (gestos).
- El software ahora acompaña al usuario 24/7 en su bolsillo.

GIT: EL NUEVO ESTÁNDAR (2005)

Linus Torvalds creó Git para el desarrollo del kernel Linux.

- Control de versiones distribuido.
- Permitió la colaboración masiva sin cuellos de botella.
- Con **GitHub** (2008), el código abierto se volvió social.

Sección IX

La Nube y el Big Data (2010 - 2020)

Infraestructura como código y contenedores.

CLOUD: EL FIN DEL SERVIDOR LOCAL

Amazon Web Services (AWS) y Azure cambiaron la escala.

- **Elasticidad:** El software puede pedir más recursos al instante.
- Pago por uso.
- Nace el concepto de **Serverless**.

DOCKER Y KUBERNETES (2013–2014)

La **Contenerización** resolvió el problema: "En mi máquina funciona".

- Empaqueta el software con todas sus dependencias.
- Kubernetes permite orquestar miles de contenedores.
- Base de la ingeniería de software escalable moderna.

DE MONOLITOS A MICROSERVICIOS

Descomponer grandes programas en piezas pequeñas independientes.

Monolito: Una sola unidad masiva → Microservicios: Decenas de servicios ágiles

Permitió a empresas como Netflix y Amazon actualizar software miles de veces al día.

FRAMEWORKS DE CIENCIA DE DATOS

El software se volvió **introspectivo**.

- Pandas, TensorFlow, PyTorch.
- El software ya no solo sigue reglas, sino que aprende de los datos.
- Sienta las bases para el boom de la IA.

Sección X

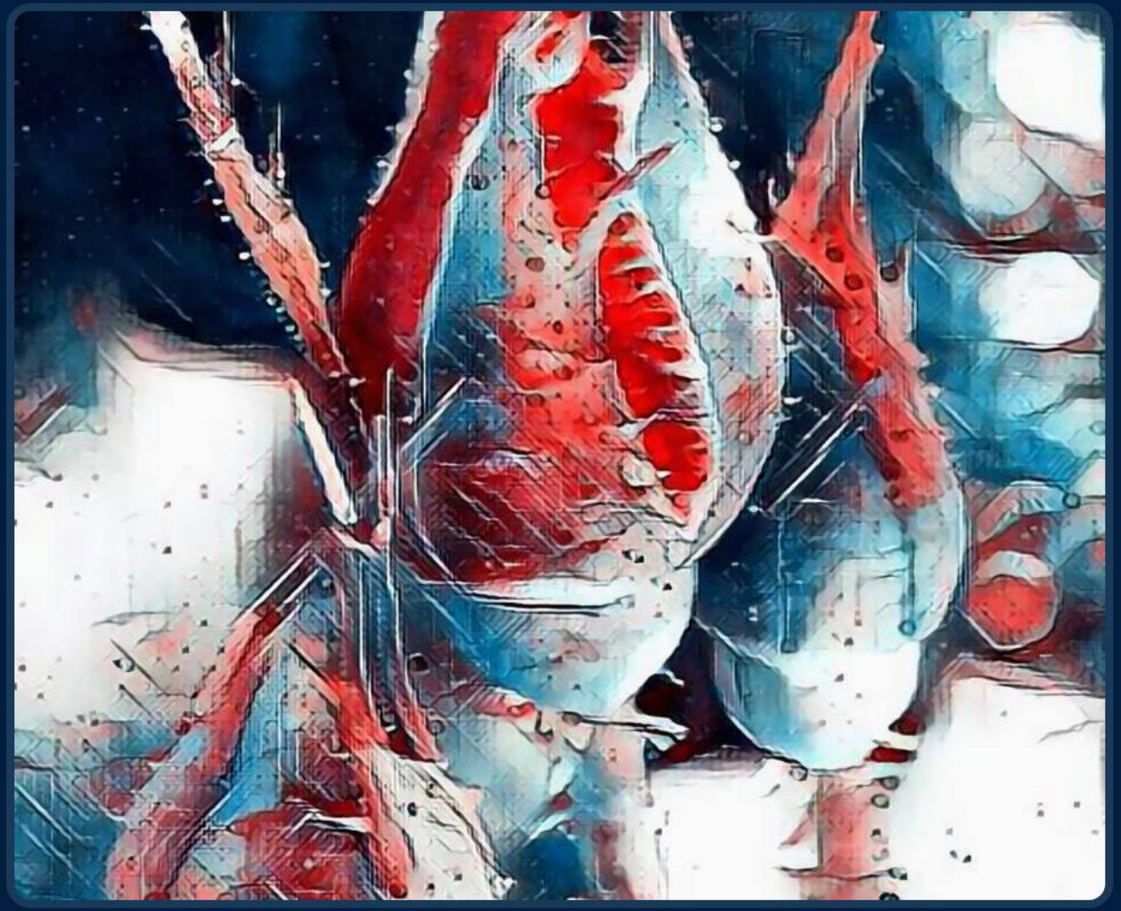
IA Generativa y el Futuro (2020 - Presente)

El software que escribe software.

LA ERA DE LOS TRANSFORMERS Y LLMS

El lanzamiento de ChatGPT (2022) marcó un cambio de era.

- Software capaz de razonar (en apariencia) y generar lenguaje natural.
- **Copilots:** La IA ahora ayuda a los humanos a escribir código.
- Programación mediante **Lenguaje Natural**.



CONCLUSIÓN: EL SOFTWARE ES EL MUNDO

De los engranajes de Babbage a las redes neuronales de hoy:

- El software ha pasado de ser una curiosidad matemática a ser el **sistema operativo de la civilización**.
- Cada avance fue posible gracias a una mayor **abstracción**.
- El futuro del software no es solo escribir código, sino resolver problemas humanos.

¡Gracias!

IMAGE SOURCES

https://upload.wikimedia.org/wikipedia/commons/thumb/a/a4/Ada_Lovelace_portrait.jpg/960px-Ada_Lovelace_portrait.jpg

Source: en.wikipedia.org

---

https://penntoday.upenn.edu/sites/default/files/styles/2880px_wide_with_focal_crop/public/2019-02/ENIACArchives.jpg?h=206f5611&itok=XFRRfe5x

Source: penntoday.upenn.edu

---

https://news.yale.edu/sites/default/files/d6_files/YaleNews_hopper-grace.UNIVAC.102635875-CC_0.jpg

Source: news.yale.edu

---

https://th-thumbnailer.cdn-si-edu.com/FWgRn5hYAspLDtzVakRB5pQxQMY=/fit-in/1600×0/filters:focal(939×447:940×448)/https://tf-cmsv2-smithsonianmag-media.s3.amazonaws.com/filer/c6/1f/c61f96b5-8da2-4715-a792-e3e6bda15714/margaret_hamilton_-_restoration.jpg

Source: www.smithsonianmag.com

---

https://www.nokia.com/sites/default/files/2025-01/ritchie-thompson-1920×1080.jpg

Source: www.nokia.com

---

https://www.techspot.com/articles-info/477/images/gui.jpg

Source: www.techspot.com

IMAGE SOURCES

https://lh3.googleusercontent.com/ci/AL18g_T7m5Ld9fGZBGbgMotYVPMjcHbkgkZZjBSPD4y0LfK0shE1UuS400xquixxTpOxaANYNMvX2w=s1200

Source: artsandculture.google.com

---

https://external-preview.redd.it/66KojmnZLWEPVblkQuQwB9z0JvJKzXhWqB4aWanNq3A.png?format=pjpg&auto=webp&s=a4688f2bdbdc10378e5edadfd88414c7639dd32c

Source: www.reddit.com

---

https://static.vecteezy.com/system/resources/previews/050/442/116/large_2x/modern-data-center-with-server-racks-and-blue-led-lights-for-cloud-computing-and-data-storage-photo.jpg

Source: www.vecteezy.com

---

https://images.saatchiart.com/saatchi/1256963/art/6199305/5269087-HSC00001-32.jpg

Source: www.saatchiart.com