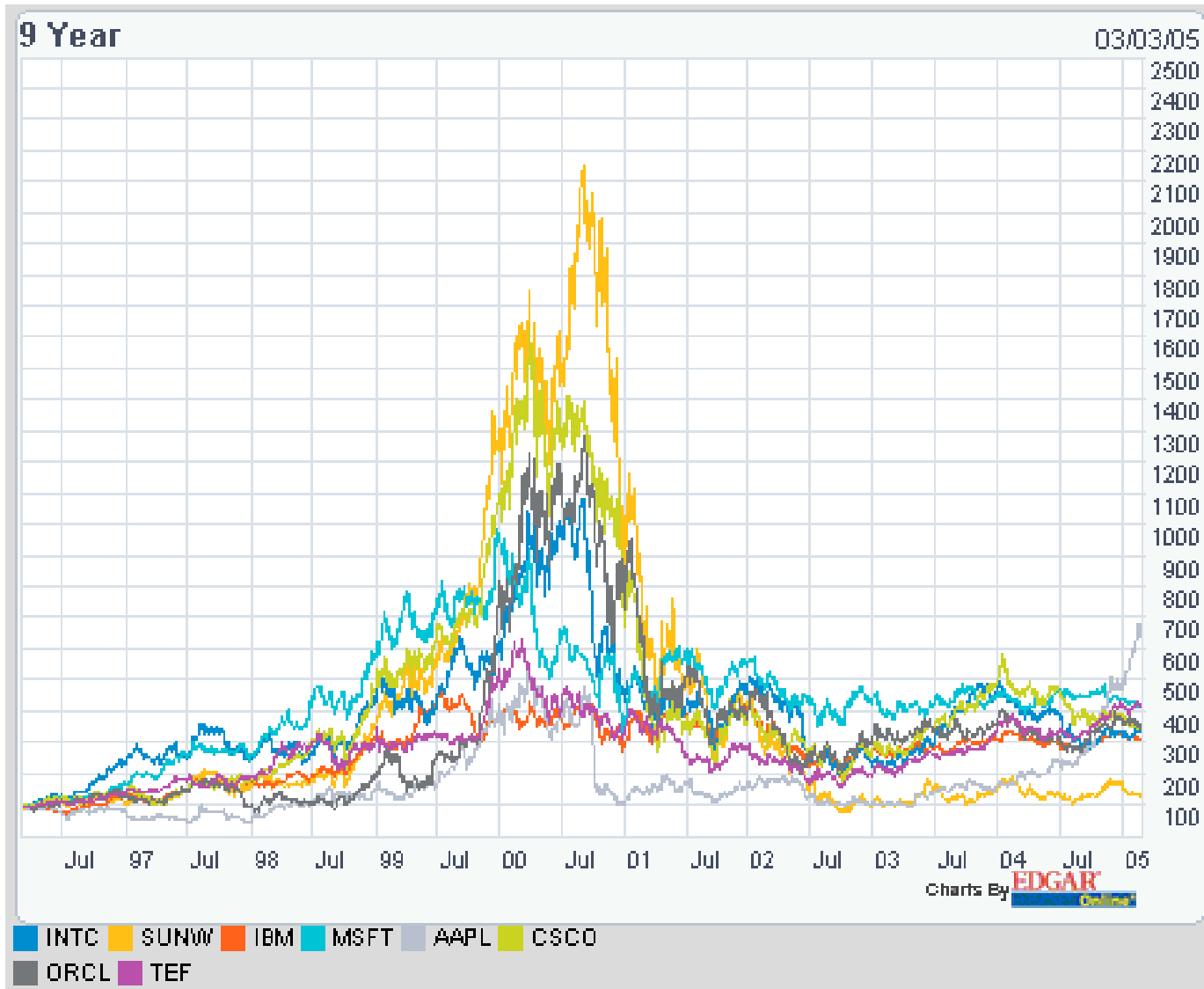


TAP. ¿ES JAVA LA MEJOR OPCIÓN?

Un componente importante del “boom” de las “dotcom” fue Java



Sun fue cada vez mejor representada por Java hasta el punto de llegar a cambiar su “ticker” en bolsa



Negocios y Mercados



Sun Microsystems pasará a ser Java en el Nasdaq

25 ago 07 | 11:25 CET

A partir del 27 de agosto, la compañía tecnológica cambiará su actual identificación bursátil, SUNW, por Java, su marca más reconocida en el mercado.

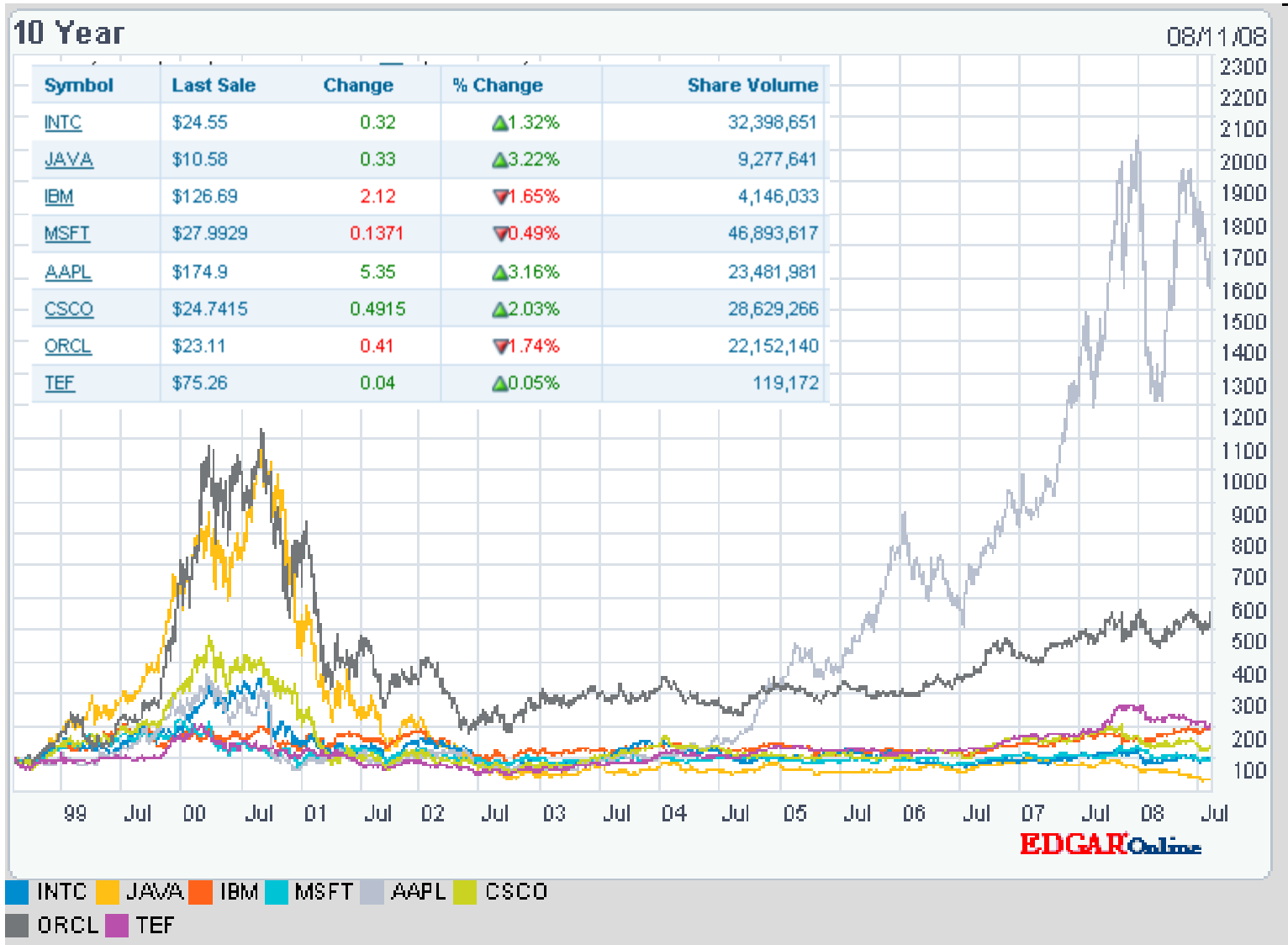
Desde el próximo 27 de agosto, la identificación bursátil de la compañía tecnológica [Sun Microsystems](#) dejará de ser SUNW para pasar a ser **Java, su marca más reconocida en el mundo del software**.

En declaraciones reproducidas por [Europa Press](#), el presidente y consejero delegado de la empresa, Jonathan Schwartz, aseguró que esta modificación de las siglas de identificación bursátil “refleja una marca que todo el mercado puede identificar y supone un elemento importante del proceso de transformación de Sun a largo plazo”.

El directivo agregó que “Java está en todas partes, tocando de cerca a cualquiera que esté relacionado con Internet y es un símbolo de la capacidad de desarrollar, introducir y dar a conocer las novedades de Sun”.

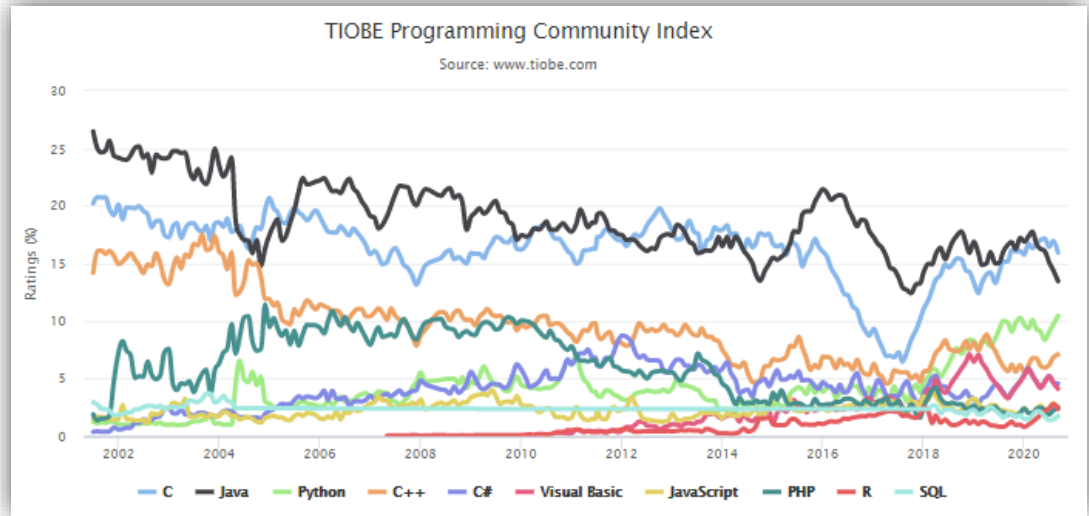
Cabe recordar que, según un informe de [Ovum](#) basado en estadísticas de mayo de este año, **existen 800 millones de ordenadores con software Java incorporado, 2.100 millones de dispositivos móviles para Java, 2.500 millones de tarjetas inteligentes y cerca de 180 operadores que ofrecen contenidos y servicios basados en esta tecnología.**

Curiosidad: Apple “relativiza” el boom de las “dotcom”



Ordenación de lenguajes por presencia en la web

Sep 2020	Sep 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	15.95%	+0.74%
2	1	▼	Java	13.48%	-3.18%
3	3		Python	10.47%	+0.59%
4	4		C++	7.11%	+1.48%
5	5		C#	4.58%	+1.18%
6	6		Visual Basic	4.12%	+0.83%
7	7		JavaScript	2.54%	+0.41%
8	9	▲	PHP	2.49%	+0.62%
9	19	▲▲	R	2.37%	+1.33%
10	8	▼	SQL	1.76%	-0.19%
11	14	▲	Go	1.46%	+0.24%
12	16	▲▲	Swift	1.38%	+0.28%
13	20	▲▲	Perl	1.30%	+0.26%
14	12	▼	Assembly language	1.30%	-0.08%
15	15		Ruby	1.24%	+0.03%
16	18	▲	MATLAB	1.10%	+0.04%
17	11	▼▼	Groovy	0.99%	-0.52%
18	33	▲▲	Rust	0.92%	+0.55%
19	10	▼▼	Objective-C	0.85%	-0.99%
20	24	▲▲	Dart	0.77%	+0.13%



Position	Programming Language	Ratings	Position	Programming Language	Ratings
21	Logo	0.76%	36	OpenEdge ABL	0.38%
22	Classic Visual Basic	0.75%	37	Fortran	0.36%
23	SAS	0.75%	38	Ada	0.34%
24	Scratch	0.73%	39	Lua	0.33%
25	PL/SQL	0.67%	40	LabVIEW	0.31%
26	Transact-SQL	0.61%	41	Prolog	0.29%
27	D	0.58%	42	TypeScript	0.28%
28	Julia	0.55%	43	VBScript	0.26%
29	Delphi/Object Pascal	0.53%	44	Haskell	0.26%
30	ABAP	0.52%	45	Hack	0.24%
31	Scala	0.52%	46	Apex	0.23%
32	Scheme	0.47%	47	PowerShell	0.23%
33	COBOL	0.43%	48	Bash	0.22%
34	Kotlin	0.41%	49	Clojure	0.22%
35	Lisp	0.40%	50	MQL4	0.21%

<https://www.tiobe.com/tiobe-index/>

¿QUÉ “CLASE” DE LENGUAJE ES JAVA?

Clasificación general de lenguajes

Euclides (Método axiomático), Aristóteles(Lógica formal), Muhammad ibn Musa Al'Khowarizmi (Algoritmo)...

1928 David Hilbert (Frege Russel Whitehead)

1931 Kurt Gödel

1938

Máquina de Turing



Alan Turing

Lenguajes



Cálculo Lambda

Alonzo Church

Sin olvidar las funciones recursivas de Herbrand-Gödel"

Imperativos

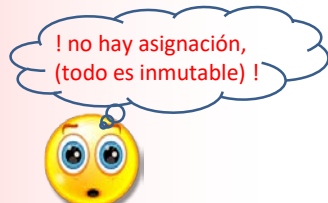
- Fortran
- Cobol
- Pascal
- C
- ...

Funcionales

- Lisp
- Scheme
- ML
- Hope
- CLOS
- Ocaml
- ...
- Haskell
- Clojure

Lógicos

- Prolog
- ...



O/B objetos

- Object Pascal
- C++
- Javascript
- **Java**
- ...

Frameworks

- **Ruby on Rails**
- ...

- Javascript
- t
- Scala
- Java 8

```
let rec long = function
  | [] -> 0
  | x::xs -> 1 + long xs;;

let rec ordenar = function
  | [] -> []
  | x::xs -> insertar x
(ordenar xs)
and insertar e = function
  | [] -> [e]
  | x::xs -> if x > e
```

Ejemplo OCaml

then e::x::xs

Alan Turing - Wikipedia, la enciclopedia libre

es.wikipedia.org/wiki/Alan_Turing ▼

Alan Mathison **Turing**, OBE (Paddington, Londres, 23 de junio de 1912 - Wilmslow, Cheshire, 7 de junio de 1954), fue un matemático, lógico, científico de la ...

Máquina de Turing

Una máquina de Turing es un dispositivo que manipula ...

Enigma

Enigma era el nombre de una máquina que disponía de un ...

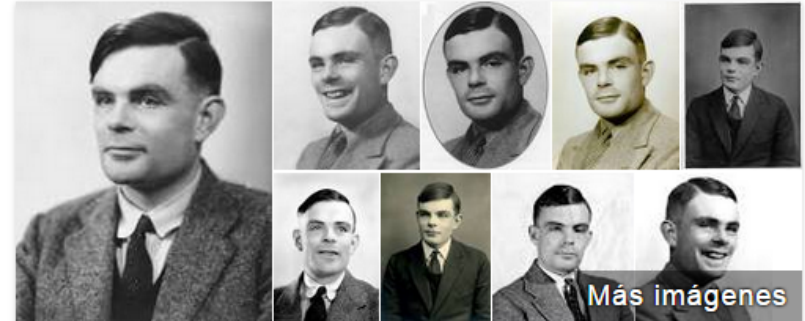
[Más resultados de wikipedia.org](#) »

Tesis de Church-Turing

En teoría de la computabilidad, la tesis de Church-Turing formula ...

Problema de la parada

El problema de la parada o problema de la detención para ...



[Más imágenes](#)

Alan Turing

Matemático

Alan Mathison Turing, OBE, fue un matemático, lógico, científico de la computación, criptógrafo y filósofo británico. Es considerado uno de los padres de la ciencia de la computación siendo el precursor de la informática moderna. [Wikipedia](#)

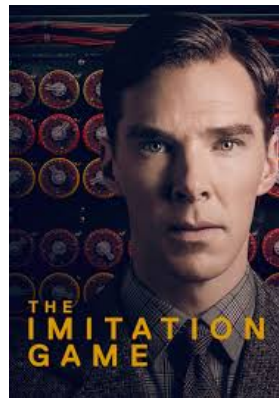
Fecha de nacimiento: 23 de junio de 1912, [Maida Vale, Londres, Reino Unido](#)

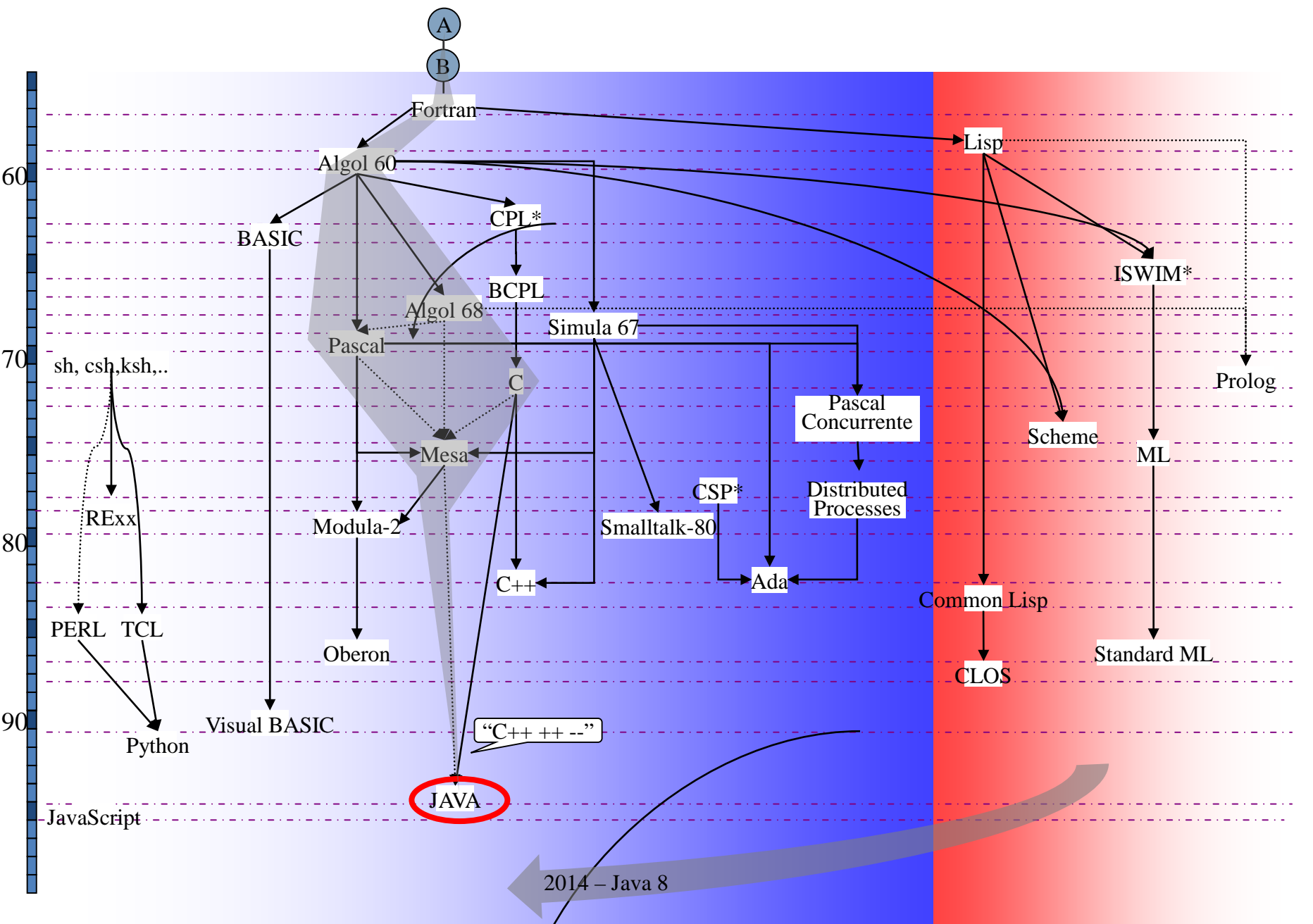
Fecha de la muerte: 7 de junio de 1954, [Wilmslow, Reino Unido](#)

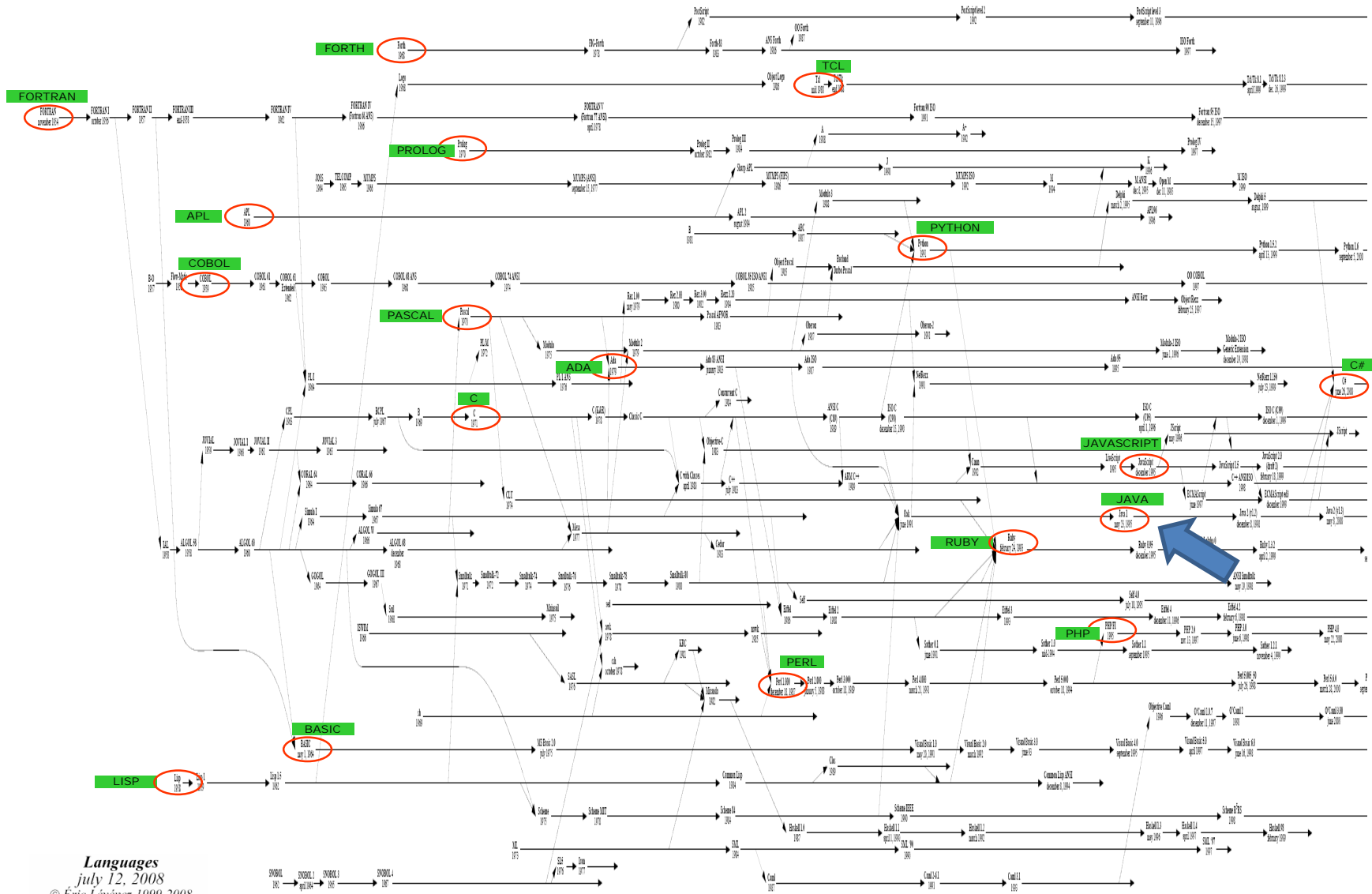
Libros: [Computing machinery and intelligence](#), [¿Puede pensar una máquina?](#)

Padres: [Julius Mathison Turing](#), [Ethel Sara Stoney](#)

Hermanos: [John Turing](#)







Languages
 July 12, 2008
 © Eric Léveñez 1999-2008
 <http://www.levenez.com/lang>

<http://www.levenez.com/lang/>

¿ES JAVA LENTO?

Es ya una cuestión muy antigua y resuelta, pero sirve para ilustrar aspectos de la ingeniería del software: programar no es sólo escribir programas

¿Java para cómputo intensivo?

(X=matemáticas, física, ingeniería...)


Tradicionalmente, en computación para X, se ha buscado la “velocidad”.

Esta visión X=cómputo intensivo puede ser cierta pero quizás parcialmente.

En todo caso MUCHÍSIMAS de las necesidades de computación “al límite” de hace unos años, hoy en día son livianas o “razonables”. (mi conjetura: El mundo de lo “intratable” por “impotencia computacional” se ha reducido enormemente)

Esta ¿obsesión?/¿necesidad? justificaba el inmovilismo de las X frente a nuevos lenguajes (debería llevarles a programar directamente los microprocesadores en su lenguaje ensamblador, pero curiosamente no se daba esto).

Resultado: FORTRAN es la referencia, y el razonable paso a C ¿se dio?.

Esta página contiene apreciaciones discutibles.
(pero las conclusiones son VERDAD) 



La Web Resultados **1 - 10** de aproximadamente **122.000** de **mathematics "in FORTRAN" program.** (0,22 segundos)

La Web Resultados **1 - 10** de aproximadamente **1.630.000** de **mathematics "in C" program.** (0,30 segundos)

(comparación inválida sin duda. La presencia “arrasadora” de C junto a la “inteligencia” de Google (que utiliza el sinónimo “math” en la búsqueda) potencian el segundo resultado.

A Java se le ha “acusado” desde un principio de ser LENTO.

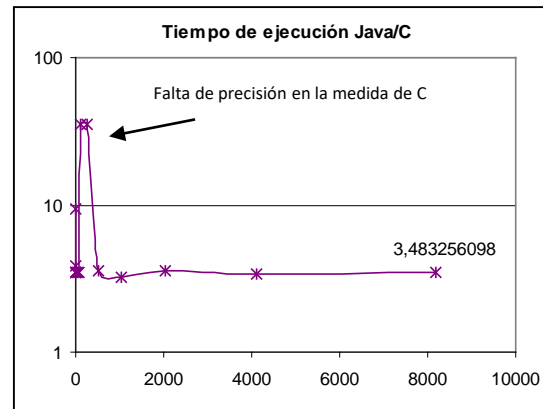
- Al principio era cierto.
 - Relación 4/1 frente a C
 - Razón principal: lenguaje interpretado
 - Otras razones: recogida de basuras, mecanismos de seguridad, etc.
- Desde hace unos años es comparable a C, dependiendo de en qué tareas. Ciertamente no es el mejor caso el del cómputo intensivo (estimaciones de un estudio particular 2004).
 - En gráficos bate a C
 - Relación media: 1.4/1 frente a C si excluimos gráficos
 - En cálculo intensivo la diferencia es más acusada
- Actualmente Java es más rápido que C en muchas tareas (particularmente gráficos) y similar en los peores casos (cómputo intensivo)
 - Máquina HotSpot
- Java (los lenguajes sobre máquinas virtuales en general) serán los más rápidos en el futuro (ante cálculos complejos, no para el caso de algoritmos muy “cerrados”)
 - El futuro de las máquinas virtuales

cómputo intensivo?... Una experiencia concreta

Ejemplo peor caso Java vs. C (14ago08)
(cálculo de PI por MonteCarlo)

Experimento a partir del código tomado de <http://husnusensoy.blogspot.com/2006/06/c-vs-java-in-number-crunching.html>

- Comparación del tiempo de ejecución



- Comparación del tiempo de preparación del experimento

JAVA:

- copiar, pegar, compilar, ejecutar y **listo en unos segundos.**

C:

- copiar, pegar, compilar, errores... (no coincide exactamente el lenguaje)
- corregir fuente, compilar, ejecutar, errores... (la arquitectura de la máquina no es la adecuada)
- corregir fuente, compilar, ejecutar, se observar falta de resolución de la función "time",
- ir a la bibliografía para resolver el tema, no encontrar solución...
- replantear con iteraciones para obtener tiempos mayores...
- cambiar fuente compilar, ejecutar... errores de apuntadores (falta de práctica de un "ex" de C)
- corregir fuente, compilar, ejecutar... errores de violación de segmentos
- corregir fuente, compilar, ejecutar y... **listo en una hora.**

La relación de tiempo de preparación ha sido de 120 a favor de Java

<http://gtts.ehu.es/German>