



INTRODUCCIÓN

LENGUAJE JAVA



INDICE

- ¿Qué es Java?
- La plataforma Java 2
- La Máquina Virtual de Java
- Características principales
- ¿Qué ventajas tengo como desarrollador?



¿Qué es Java?

La tecnología Java es un lenguaje de programación y una plataforma.



La plataforma Java 2

- Una plataforma software que se ejecuta sobre otra plataforma hardware/software.

- La Máquina Virtual Java
(JVM)

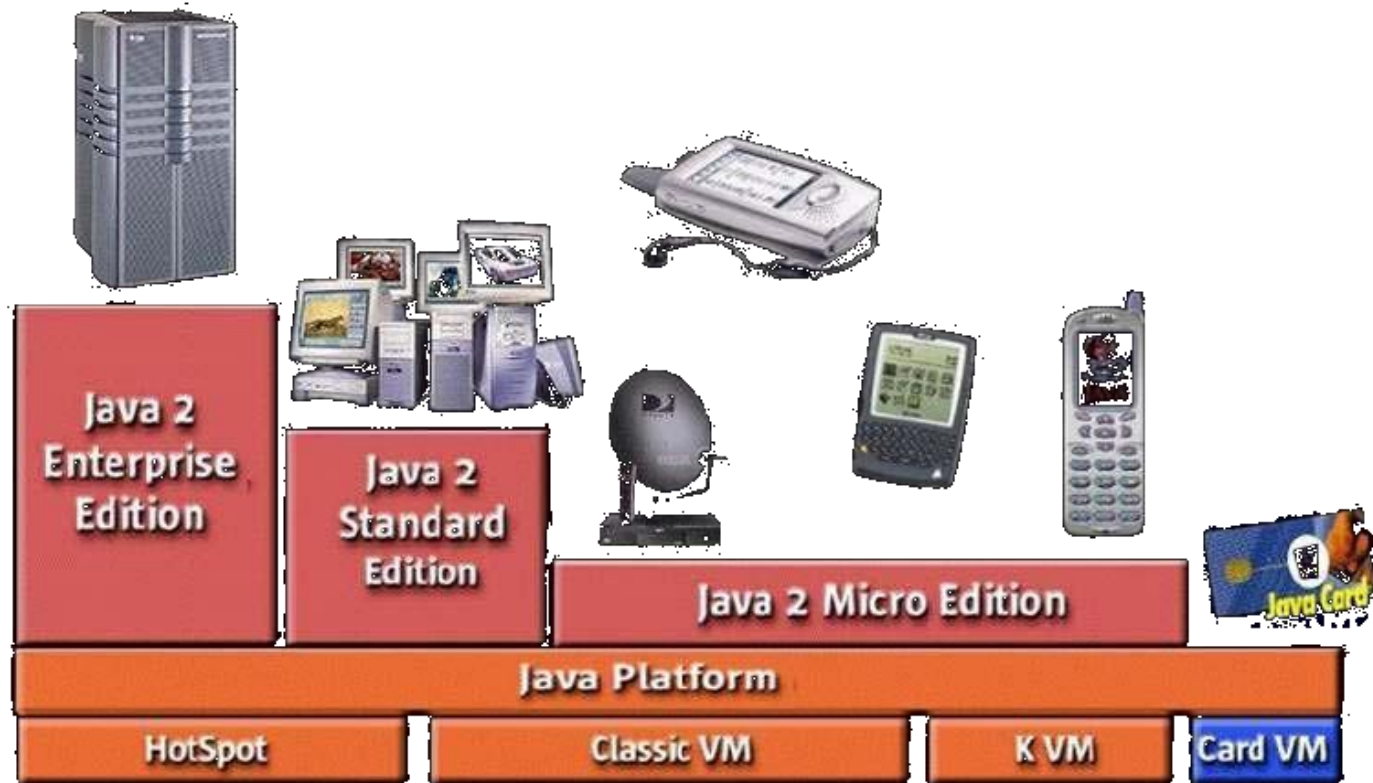
Intérprete de Java

La interfaz de Programación
de Aplicaciones (API)

Conjunto de clases ya desarrolladas



La plataforma Java 2



Características Principales

- Sencillo
- Orientado a objetos
- Distribuido
- Interpretado
- Robusto
- Seguro
- Arquitectura neutra
- Portable
- Altas prestaciones
- Multithread (multihilo)
- Dinámico



Características: Sencillo

- Elimina algunas de las características de C y C++:
 - No aritmética de punteros.
 - No existen referencias.
 - No existe typedef ni #define.
 - No sobrecarga operadores.
 - No herencia múltiple.
 - Posee clase String.
 - No necesidad de asignar y liberar memoria (Garbage Collector).

Características: Orientado a objetos

- Java trabaja con sus datos como objetos y con interfaces a esos objetos.
- Implementa:
 - Encapsulación.
 - Herencia simple.
 - Abstracción.
 - Reutilización.
 - Polimorfismo.
 - Interfaces.
 - Paquetes.



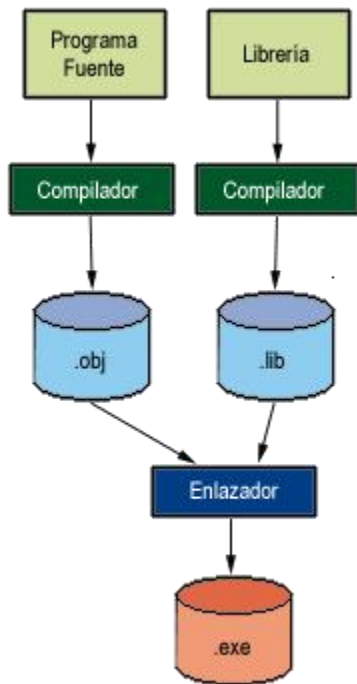
Características: Distribuido

- Concebido para trabajar en un entorno conectado en red.
- En sí no es distribuido, proporciona librerías y herramientas para que los programas puedan ser distribuidos, que los programas corran en varias máquinas interactuando.



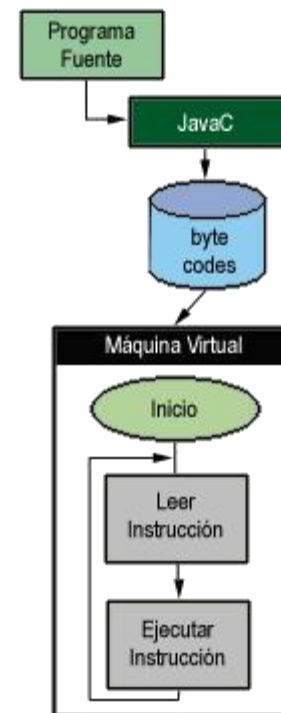
Características: Interpretado

- Compilador tradicional



- » Traslada sentencias alto-nivel a múltiples instrucciones.
- » Creación de librerías como resultado de compilaciones previas.
- » Juntando, se crea programa ejecutable.

- Intérprete de ByteCodes



» Compilador Java

Traduce código fuente a código intermedio.

» Interprete de Java

Virtual Machine(JVM)
(ByteCodes interpretados (ejecutados))

Características: Robusto

- No se interrumpe fácilmente.
- Evita sobrescribir o corromper memoria mediante punteros.
- Manejo de excepciones.
- Verificación de ByteCodes (por ejemplo no acceso ilegal de memoria).



Características: Seguro

- Aplicaciones seguras ya que no acceden a zonas delicadas de memoria o de sistema
 - No existen punteros, no acceso a zonas delicadas de memoria.
 - Una applet no puede acceder al sistema, a los archivos de la persona que lo está viendo.



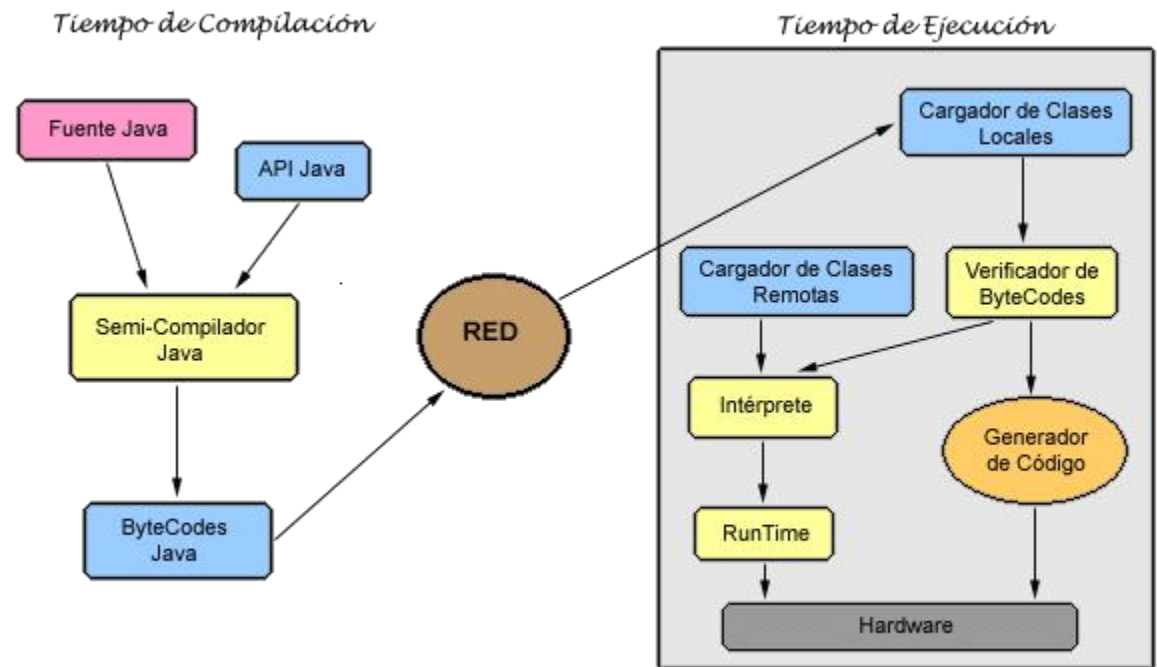
Características: Arquitectura neutra

- Java compila el código a un fichero objeto de formato independiente de la arquitectura de la máquina en que se ejecutará.
- Cualquier plataforma donde exista una JVM se puede ejecutar el código objeto independientemente de la máquina en el que ha sido generado.



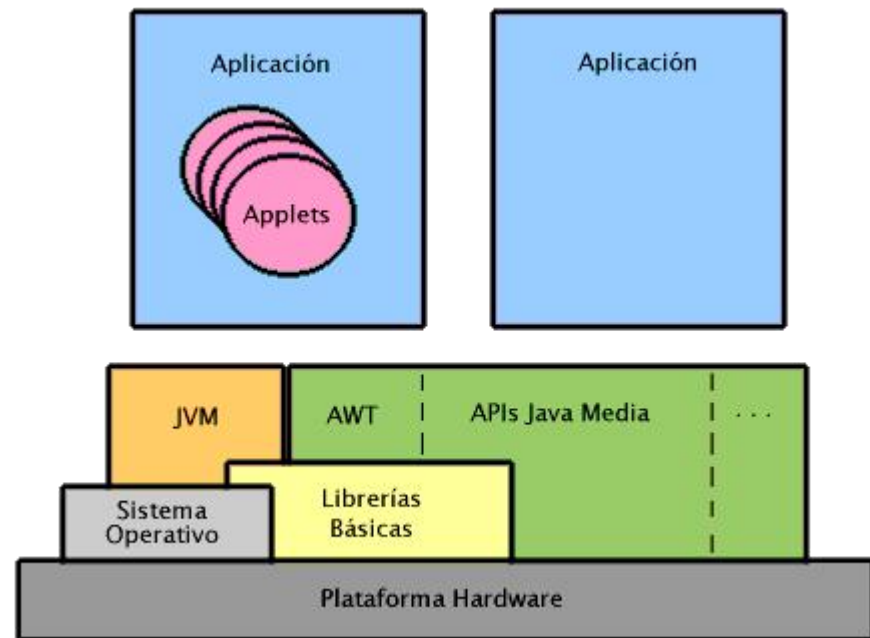
Características: Arquitectura neutra

- Código fuente se compila a un código de bytes independientemente de la máquina.
- Los ByteCodes se ejecutan en una máquina hipotética que se implementa por un sistema run-time, que es dependiente de la máquina.



Características: Arquitectura neutra

- Elementos que forman parte de arquitectura de Java sobre una plataforma genérica.
- Dependiente del sistema:
 - JVM.
 - Librerías fundamentales.
 - APIs de Java que entren en contacto directo con el hardware.



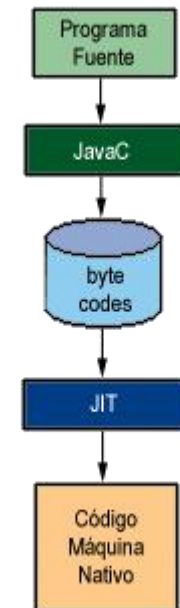
Características: Portable

- Más allá de la portabilidad básica por ser de arquitectura independiente.



Características: Altas prestaciones

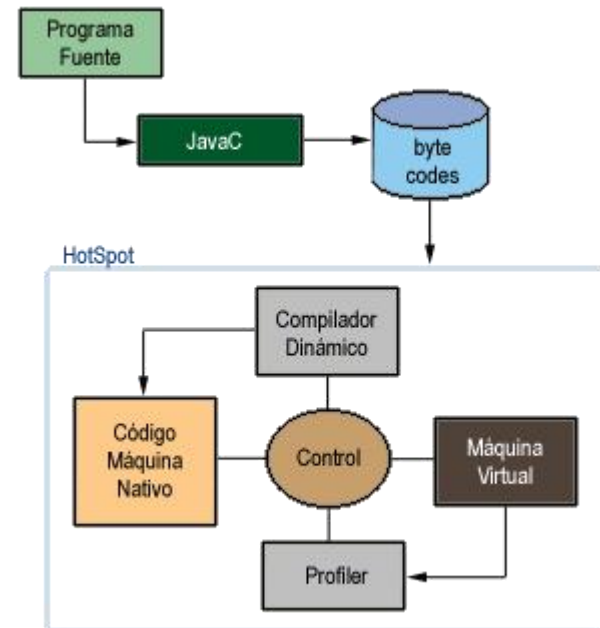
- Compiladores JIT (Just in time): Compilan en el momento de ejecución.
 - Traslada los bytecodes al código máquina de la plataforma según los va leyendo, realizando cierto grado de optimización.
 - JIT no pierde tiempo en optimizar código que no se ejecutará.



Características: Altas prestaciones

- Se introdujo un nuevo JIT: HotSpot.

- Incluye compilador dinámico y una máquina virtual para Interpretar bytecodes.
- Mejor soporte para ordenadores con varios procesadores, mayor escalabilidad.



Características: Multithread

- Ejecución de varias tareas a la vez. Permite muchas actividades simultáneas en un programa.
- Posee una serie de clases que facilitan su utilización.
- Mejor rendimiento interactivo y mejor comportamiento en tiempo real.

Características: Dinámico

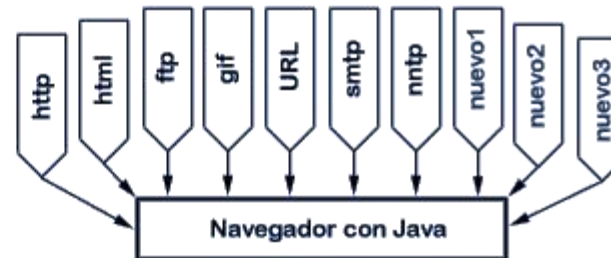
- No conecta todos los módulos que comprenden una aplicación hasta el mismo tiempo de ejecución.
- Las librerías nuevas o actualizadas no paralizan la ejecución de las aplicaciones siempre que mantengan la API anterior.

Características: Dinámico

- Capaz de traer automáticamente cualquier pieza que el sistema necesite para funcionar.



Monolito: cada pieza de código se compacta dentro del código del navegador



Sistema Federado: el navegador es un coordinador de piezas, y cada pieza es responsable de una función. Las piezas se pueden añadir dinámicamente a través de la red

¿Qué ventajas tengo como desarrollador?

- Comenzar rápidamente.
- Escribir menos código, 4 veces menos que en C++.
- Escribir mejor código.
- Desarrollar programas más rápidamente.
- Eliminar dependencias de la plataforma.
- Escribir una vez, ejecutar en cualquier parte.

Centro TICS



8/2/2022

Centro TICS