

El Lenguaje y la Plataforma JAVA

La Plataforma abarca dos aspectos:

- Una Plataforma de Software
- Un Lenguaje de Programación

La Plataforma JAVA

Oracle provee dos versiones de la Plataforma Estándar Java (Java SE): Java SE Development Kit (JDK) o **plataforma de desarrollo** y Java SE Runtime Environment (JRE) o **plataforma de ejecución**.

La última versión de esta plataforma es la versión 14.

Plataforma/Producto que la implementa	Nombre completo	Abreviatura
Nombre de la Edición de la Plataforma	Java Platform, Standard Edition 14	Java SE14
Producto implementa plataforma de desarrollo	Java SE Development Kit 14	JDK 14
Producto implementa plataforma de ejecución	Java SE Runtime Environment 14	JRE 14

La Plataforma de ejecución JAVA

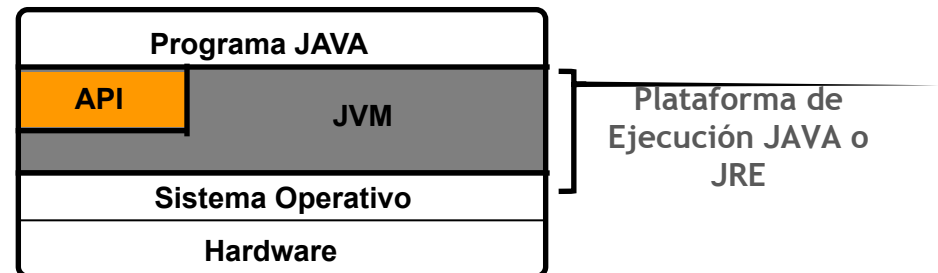
Una **plataforma de ejecución** es la combinación de un hardware y un sistema operativo que provee los servicios necesarios para ejecutar programas. Ejemplos de plataformas de ejecución son: Linux sobre una computadora Intel I7, Solaris ejecutándose sobre hardware Sun, etc.

La **plataforma de ejecución JAVA** provee los servicios necesarios para ejecutar programas escritos en JAVA. La plataforma JAVA se ejecuta sobre cualquier plataforma de ejecución.

A la plataforma de ejecución JAVA se la denomina JRE (Java Runtime Environment) o Entorno de Ejecución JAVA.

La plataforma de ejecución JAVA está compuesta por una **Máquina Virtual JAVA** o **JVM** (Java Virtual Machine) y un conjunto de librerías de código JAVA compilado, comúnmente denominado Interface de Programación de Aplicaciones JAVA o API (del inglés Application Programming Interface). La JVM es parte del JRE.

Los programas escritos en JAVA se ejecutan sobre la plataforma de ejecución JAVA.



Repaso: compilar e interpretar

El **lenguaje de máquina** o **código binario** consiste de instrucciones muy simples que la CPU de la computadora ejecuta directamente. Cada tipo de procesador tiene su propio lenguaje de máquina.

Los programas se escriben en **lenguajes de programación de alto nivel**, como Java, C++, Delphi, Python, etc. Un programa escrito en un lenguaje de alto nivel no puede ejecutarse directamente en una computadora, necesita ser traducido al lenguaje de máquina de la computadora donde se ejecutará. Este proceso de traducción lo realiza un programa llamado *compilador*.

El programa compilado también llamado código binario, puede ejecutarse tantas veces como se desee. Típicamente, el código ejecutable es particular de la plataforma de ejecución donde se ejecutará el programa; es dependiente de la plataforma de ejecución.

Una alternativa a compilar un programa escrito en un lenguaje de alto nivel, es interpretarlo: un *intérprete* es un programa que traduce y ejecuta un programa escrito en un lenguaje de alto nivel, instrucción por instrucción. A diferencia del compilador, el intérprete no genera un código ejecutable, sino que traduce a código binario y ejecuta, de a una línea por vez, cada vez que se ejecuta el programa.

La Máquina Virtual

La **Máquina Virtual Java** o **JVM** es software y constituye el corazón del JRE.

El lenguaje de máquina de la Máquina Virtual Java es la **codificación de bytes** o **bytecodes**. Es un lenguaje intermedio entre la codificación binaria específica del procesador y el lenguaje de alto nivel.

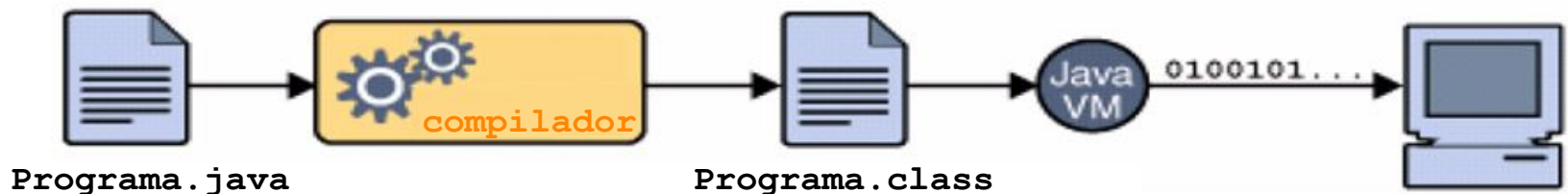
Una de las componentes fundamentales de la JVM es el intérprete JAVA, su responsabilidad es traducir *bytecodes* a código binario específico y luego ejecutarlo.

Como ya vimos, el código ejecutable es dependiente de la plataforma de ejecución particular -> necesitamos una JVM particular para cada tipo de plataforma de ejecución. Una máquina virtual para Windows, para Linux, para Solaris, etc.

La JVM es un mediador entre los programas JAVA compilados y la plataforma de ejecución específica de la computadora. Sus principales funciones son: traducir y ejecutar bytecodes, administrar la memoria del sistema de ejecución, proveer un sistema de seguridad y balancear la ejecución de múltiples threads.

Los Programas JAVA son de Plataforma Neutral

En JAVA el código fuente tiene extensión *.java*. Los archivos fuentes *.java* se compilan al lenguaje de máquina de la máquina virtual JAVA (JVM). Los archivos compilados tienen extensión *.class*. Un archivo *.class* no contiene código binario para un procesador específico, sino que contiene *bytecodes*, que es la codificación que entiende la máquina virtual de JAVA.



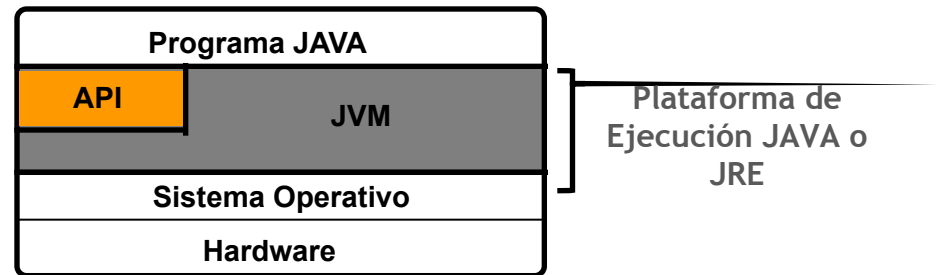
- JAVA es un lenguaje *compilado e interpretado*.
- Para ejecutar un programa JAVA compilado (.class) en una computadora SPARC con Solaris es necesario disponer de la JVM para Solaris, si lo vamos a ejecutar en una computadora Intel con Windows necesitamos la JVM para Windows. El mismo programa JAVA compilado a bytecodes lo podemos ejecutar en diferentes computadoras, no es necesario volver a compilarlo.
- La JVM garantiza que todo programa JAVA es de plataforma neutral.

Más sobre la máquina virtual

Característica principal

- Aislar al programa Java del sistema operativo y del hardware sobre el que se está ejecutando => independencia de la plataforma de ejecución.

Los programas escritos en JAVA se ejecutan sobre la plataforma de ejecución JAVA.

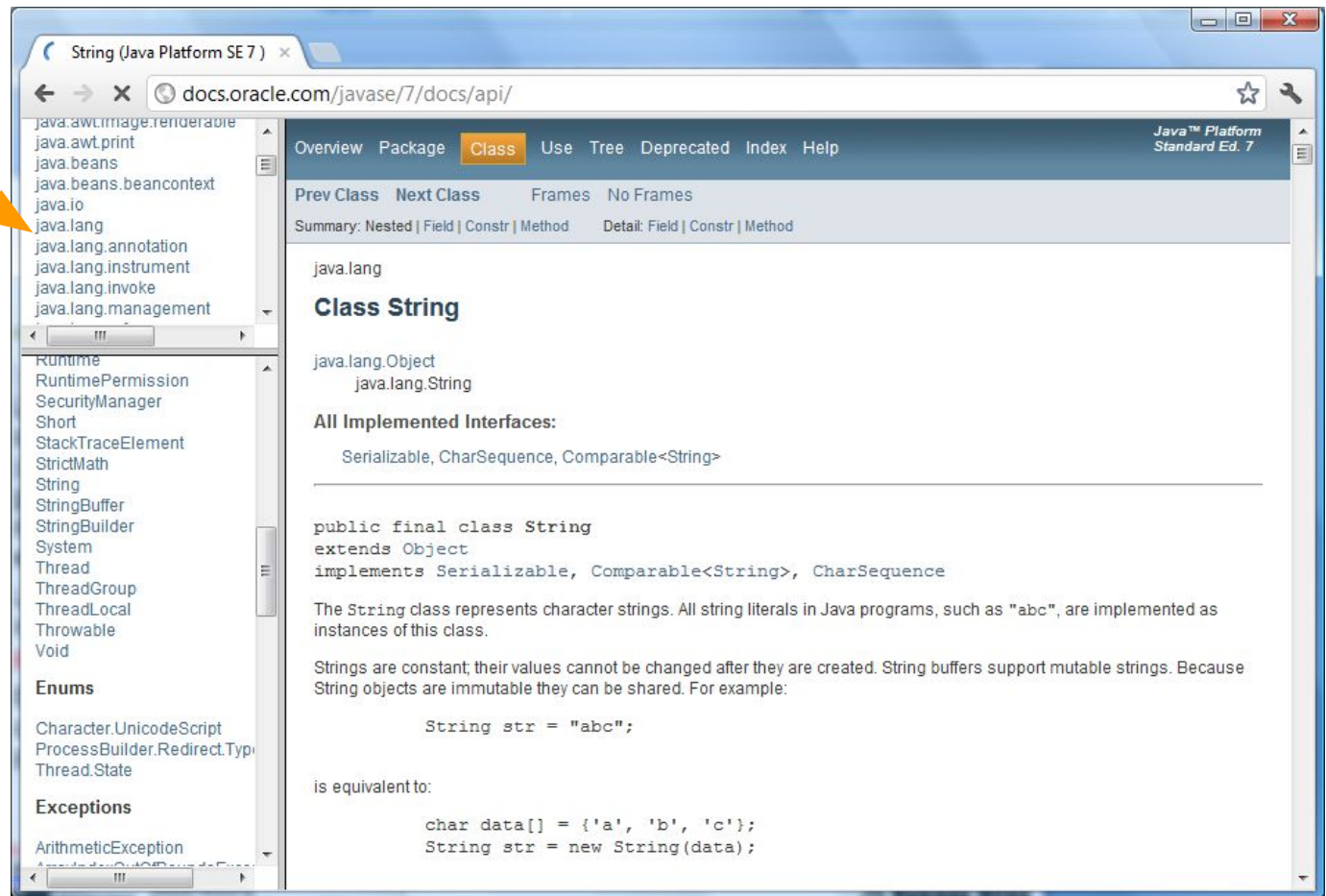


- La especificación de la JVM es única y permite que los programas Java sean independientes de la plataforma de ejecución ya que se compilan para una máquina genérica, la JVM, y se ejecutan en cualquier computadora que disponga de la JVM.
- La JVM asegura la portabilidad de los programas Java.
- La especificación de la JVM es un estándar. Cada sistema operativo tiene su propia implementación de la JVM.

API (Application Programming Interface)

La API JAVA es una colección de clases y otras componentes de software compiladas (archivos .class) que proveen una amplia gama de funcionalidades como componentes de GUIs, I/O, manipulación de colecciones, etc.

La API está agrupada en librerías de clases e interfaces Java relacionadas, llamadas paquetes. El programador puede combinar las componentes de la API JAVA con su código para crear una aplicación.



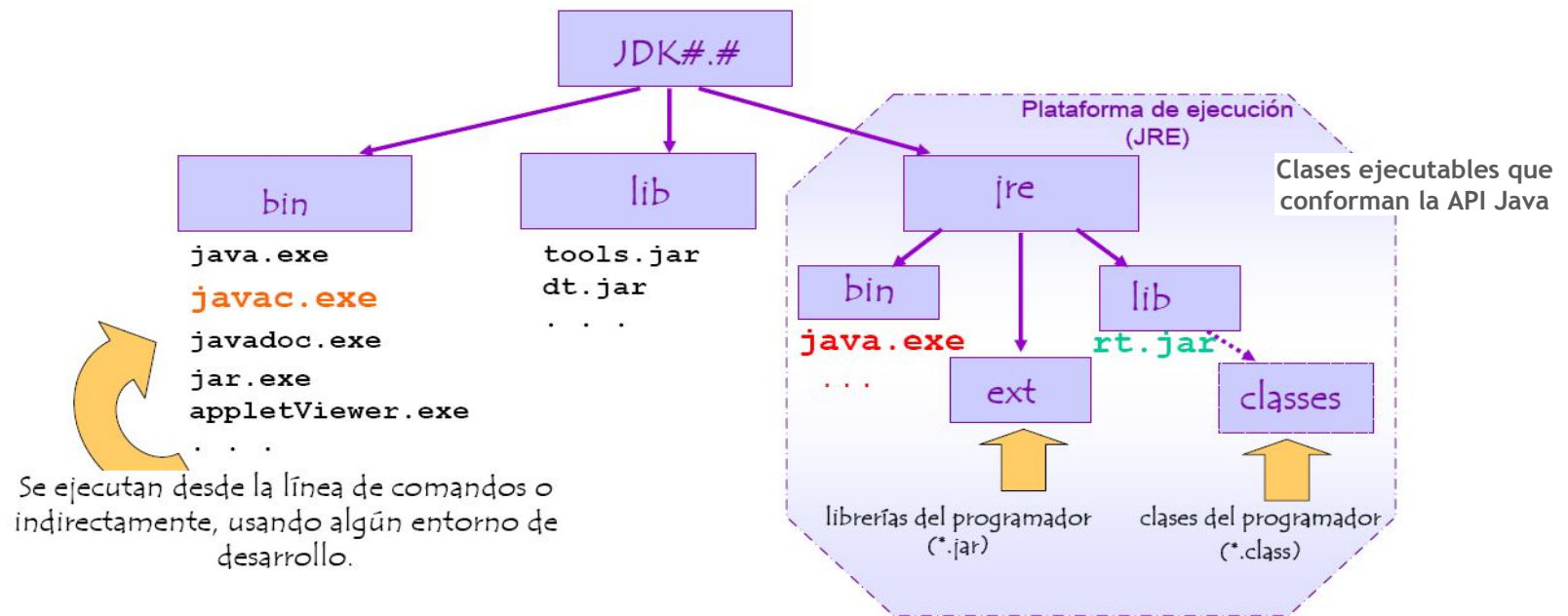
API (Application Programming Interface)

Algunos paquetes de la API JAVA son:

- *java.lang*: contiene clases esenciales como números, strings, objetos, seguridad y threads. Es el único paquete que se incluye automáticamente en todo programa Java.
- *java.io*: contiene las clases que manejan la Entrada/Salida, Serialización de objetos.
- *java.util*: contiene clases útiles que permiten manejar estructuras de datos o colecciones, fechas, hora, etc.
- *java.net*: contiene clases como URL, TCP, UDP, IP, etc. que permiten implementar aplicaciones distribuidas. Provee soporte para sockets.
- *java.awt*: contiene clases para el manejo de la GUI, pintar gráficos e imágenes.
- *java.applet*: contiene clases útiles para la creación y manipulación de applets y recursos para reproducción de audio.
- *java.sql*: contiene clases para el manejo de base de datos relaciones.

La plataforma de desarrollo

La plataforma de desarrollo JAVA se conoce como **JDK (Java Development Kit)**. Es un super-conjunto del JRE y contiene todo lo que está en el JRE más herramientas de desarrollo como un *compilador*, *debugger*, *compactador*, *documentador*, etc.; necesarios para desarrollar applets y aplicaciones de escritorio. También dispone de herramientas de seguridad y de despliegue de aplicaciones. La estructura de directorios del JDK es la siguiente:



Ediciones de la Plataforma Java

- **JSE (Java Standard Edition)**: está diseñada para programar y ejecutar applets y aplicaciones de escritorio JAVA. Típicamente son programas que se ejecutan en una PC. Es el fundamento de las 2 restantes ediciones. Está compuesta por el JRE y el JDK.
- **JEE (Java Enterprise Edition)**: está diseñada para programar y ejecutar aplicaciones empresariales, caracterizadas por ser multiusuario y distribuidas. El procesamiento de estas aplicaciones se realiza en un servidor. Usualmente son aplicaciones web. Ejemplos de este tipo de aplicaciones son: aplicaciones bancarias, de administración de recursos humanos en organizaciones gubernamentales, sistemas de hospitales, etc.
- **JME (Java Micro Edition)**: está diseñada para programar y ejecutar aplicaciones para dispositivos con recursos de cómputo limitados, como pueden ser teléfonos celulares, palms, pdas, etc. Estos dispositivos cuentan con poca memoria RAM, pantallas muy chicas inclusive algunos carecen de ellas, la conexión de red puede ser intermitente, etc.



Java es estándar

- Todas las tecnologías Java tienen una especificación desarrollada por el Java Community Process (JCP): <http://jcp.org/en/home/index>
- El Java Community Process es la comunidad de desarrollo de especificaciones de tecnologías JAVA, liderada por SUN Microsystem.
- Contar con una especificación, transforma a todas las tecnologías JAVA en estándares. Cada fabricante de software desarrolla su implementación respetando la especificación. De esta manera se garantiza compatibilidad y portabilidad.
- Las especificaciones para JSE, JEE y JME junto con las APIs asociadas son desarrolladas por el JCP.
- Una especificación comienza como un Java Specification Request (JSR) que pasa por varios estados en la JCP antes de convertirse en la especificación definitiva. Cada JSR tiene asignado un número:

JSR 386: Especificación de Java SE 12

JSR 366: Especificación de Java EE 8

. . .

Java es estándar

Existen múltiples Entornos de Desarrollo o IDEs para trabajar con la plataforma Java SE y de esta manera facilitan el desarrollo de aplicaciones Java. Proveen un conjunto de herramientas como editores contextuales, visuales, refactoring, paletas de componentes de interfaz de usuario, consola de salidas, etc.

IDE	Software Libre y Código Fuente Abierto	Propietario
Eclipse	✓	
NetBeans	✓	
IBM Rational Application Developer (RAD)		✓
Oracle JDeveloper		✓
IntelliJ IDEA	✓	✓

Rational
the software development company



netBeans.ORG

ORACLE
JDEVELOPER



Características del Lenguaje

- **Simple y Familiar:** JAVA tiene un núcleo consistente de conceptos fundamentales que facilita su aprendizaje. La sintaxis de JAVA es similar a la C++, aunque de menor complejidad.
- **Orientado a Objetos:** los programas JAVA están basados en objetos y en las interfaces para trabajar con esos objetos.
- **Robusto:** los programas JAVA son chequeados estrictamente mediante software antes de ejecutarse.
- **Seguro:** JAVA incorpora características de seguridad para garantizar que los programas que se ejecutan en una red no dañen los archivos de la computadora ni introduzcan virus.
- **Portable:** los programas JAVA pueden fácilmente ser transferidos de plataforma. Por ej. pueden pasar de ejecutarse en una PC con Windows a un servidor con Linux.
- **Alta performance:** los programas Java pueden ejecutarse eficientemente.
- **Multihilo:** soporta *threads* lo que permiten hacer múltiples tareas simultáneas logrando mejor tiempo de respuesta y comportamiento de los programas.
- **Dinámico:** los programas JAVA pueden adaptarse en ejecución a los cambios.

Referencias

Documentación de la API de Java:

<http://docs.oracle.com/javase/7/docs/api/>

Descarga de la plataforma JAVA:

<https://www.oracle.com/java/technologies/javase-downloads.html>