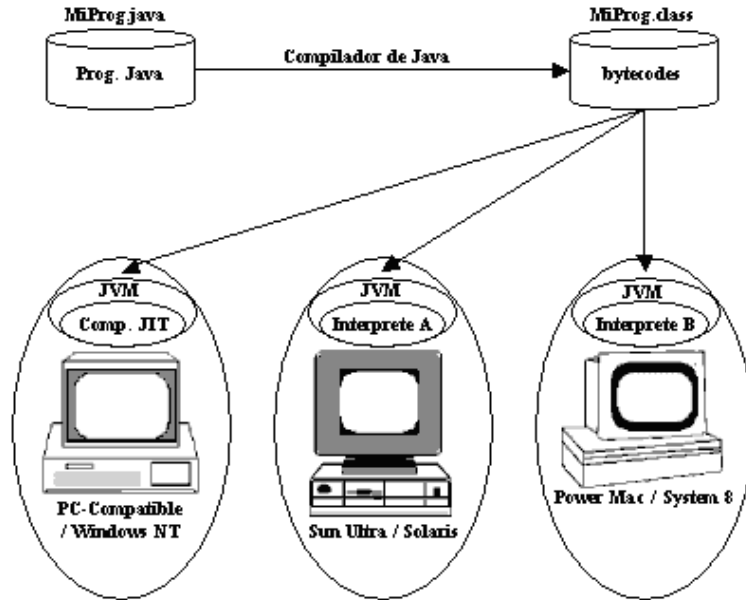


LA MÁQUINA VIRTUAL

"WRITE ONCE, RUN ANYWHERE"



Proprietary/closed source implementations

- * Hewlett-Packard's Java for HP-UX, OpenVMS, Tru64 and Reliant (Tandem) UNIX platforms
- * J9 VM from IBM, for AIX, Linux, MVS, OS/400, Pocket PC, z/OS
- * Mac OS Runtime for Java (MRJ) from Apple Inc.
- * JRockit from BEA Systems acquired by Oracle Corporation
- * Oracle JVM (also known as "JServer" and as "OJVM") from Oracle Corporation
- * Microsoft Java Virtual Machine (MS JVM) from Microsoft
- * PERC from Aonix is a real time Java for embedded
- * JBed from Esmertec is an embedded Java with multimedia capabilities
- * JBlend from Aplix is a Java ME implementation
- * Excelsior JET (with AOT compiler)

Lesser-known proprietary JVMs

- * Blackdown Java (port of Sun JVM)
- * CVM
- * Gemstone Gemfire JVM - modified for J2EE features
- * Golden Code Development (EComStation and OS/2 port of Java RTE and SDK for J2SE v1.4.1_07)
- * Tao Group's intent
- * Novell, Inc.
- * NSIcom CrE-ME
- * HP ChaiVM and MicrochaiVM
- * MicroJVM from Industrial Software Technology (running of wide range of microcontrollers 8/16/32-bit)

Free/open source implementations

- | | | | |
|------------------|-------------|-------------------------|---------------|
| * AegisVM | * JamVM | * Juice | * Mika VM |
| * Apache Harmony | * Jaos * | * Jupiter JVM | * Mysaifu JVM |
| * CACAO | JC | * JX (operating system) | * NanoVM |
| * IcedTea | * Jikes RVM | * Kaffe | * SableVM |
| * IKVM.NET | * JNode | * leJOS | * SuperWaba |
| * Jamiga | * JOP | | * TinyVM |
- * JESSICA (Java-Enabled Single-System-Image Computing Architecture)
- * Squawk virtual machine (Sun JVM for embedded system and small devices)
- * Sun Microsystems' HotSpot
- * VMkit of Low Level Virtual Machine
- * Wonka VM
- * Xam



- Una idea novedosa, pero no del todo: cierta similitud con los lenguajes con código intermedio.
- Sí fue novedoso el enfoque de emulador de máquina (y la compilación JIT).
- Ventajas:
 - se pueden incluir con facilidad técnicas que en un diseño hardware pueden resultar prohibitivas por su complejidad técnica,
 - la posibilidad de evolución es mucho más sencilla al no requerir cambios de hardware
 - permite utilizar las "plataformas" existentes sin implicar una ruptura con los sistemas actuales (existe la máquina real pero...).
- el diseño es público y la "implementación" es privada ([especificaciones técnicas que debe cumplir toda JVM.](#)).
- Distintos comportamientos en términos de velocidad y uso de memoria

Comentario 21/9/18:
La que han liado
"los de Python"

En la máquina Virtual está el “secreto”...

además aporta otras muchas ventajas.

Cambios importantes en la evolución de Java:

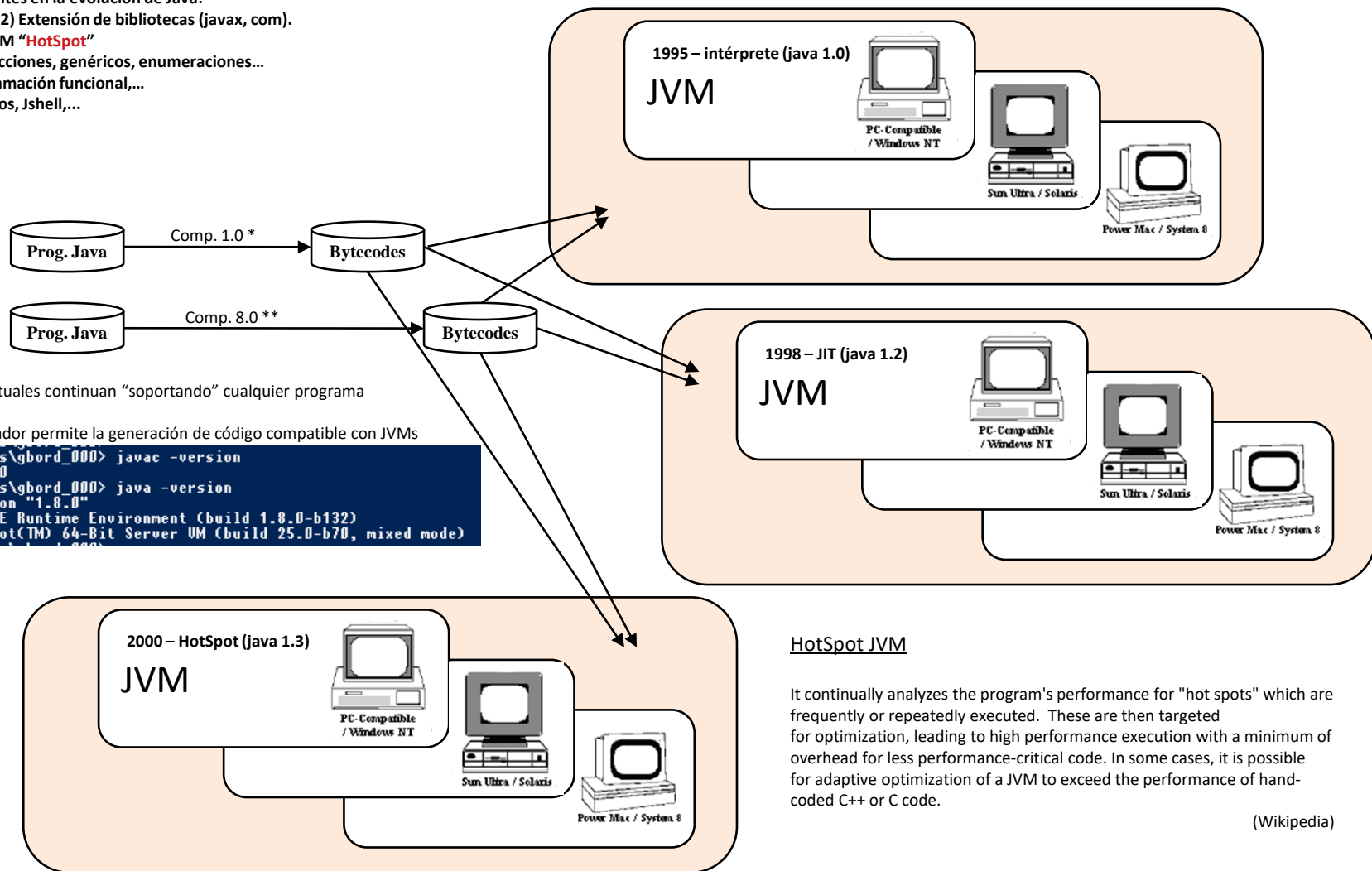
Java 1.2 (Java2) Extensión de bibliotecas (javax, com).

Java 1.3 La JVM “HotSpot”

Java 1.5: colecciones, genéricos, enumeraciones...

Java 8 Programación funcional,...

Java 9 Módulos, Jshell,...



* Las JVM actuales continúan “soportando” cualquier programa antiguo.

** El compilador permite la generación de código compatible con JVMs

```
PS C:\Users\gbord_000> javac -version
javac 1.8.0
PS C:\Users\gbord_000> java -version
java version "1.8.0"
Java(TM) SE Runtime Environment (build 1.8.0-b132)
Java HotSpot(TM) 64-Bit Server VM (build 25.0-b70, mixed mode)
```

HotSpot JVM

It continually analyzes the program's performance for "hot spots" which are frequently or repeatedly executed. These are then targeted for optimization, leading to high performance execution with a minimum of overhead for less performance-critical code. In some cases, it is possible for adaptive optimization of a JVM to exceed the performance of hand-coded C++ or C code.

(Wikipedia)

En la máquina Virtual está el “secreto”...

y no debemos confundirla con el lenguaje.

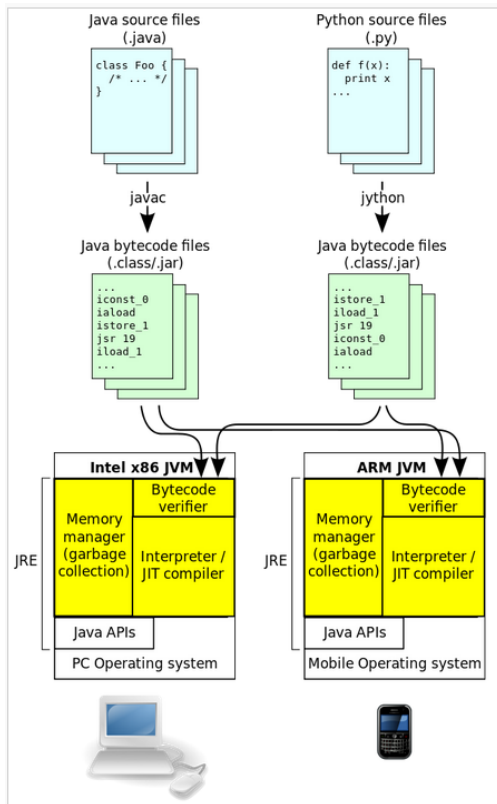
Versions of non-JVM languages

Language	On JVM
Erlang	Erjang
JavaScript	Rhino
Pascal	Free Pascal
PHP	Quercus
Python	Jython
REXX	NetRexx ^[3]
Ruby	JRuby
Tcl	Jacl

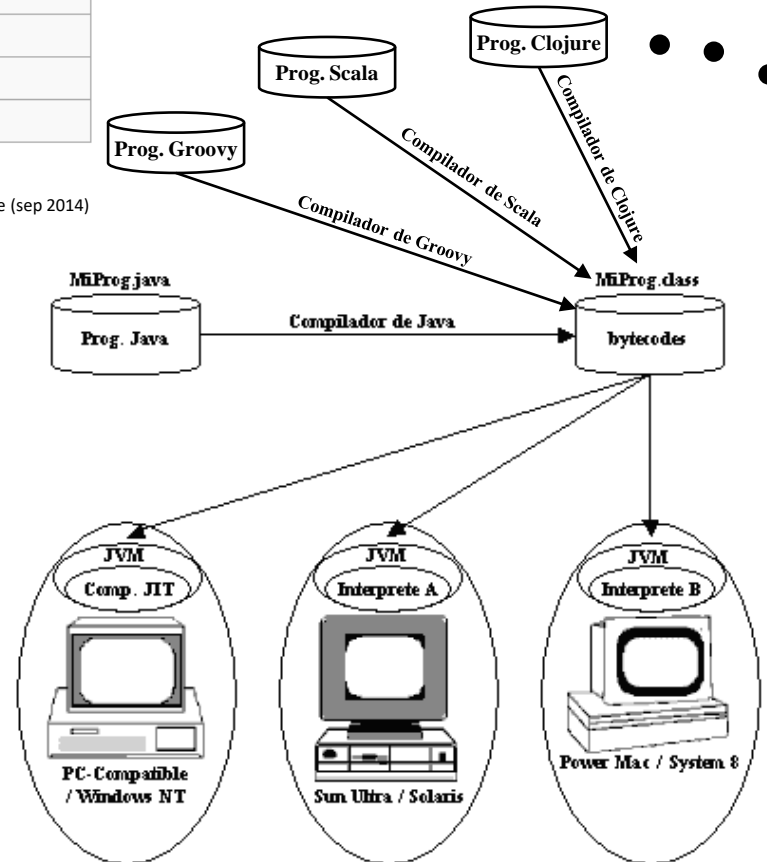
Languages designed expressly for JVM

Language
BBj
Clojure
Fantom
Groovy
MIDletPascal
Scala
Kawa

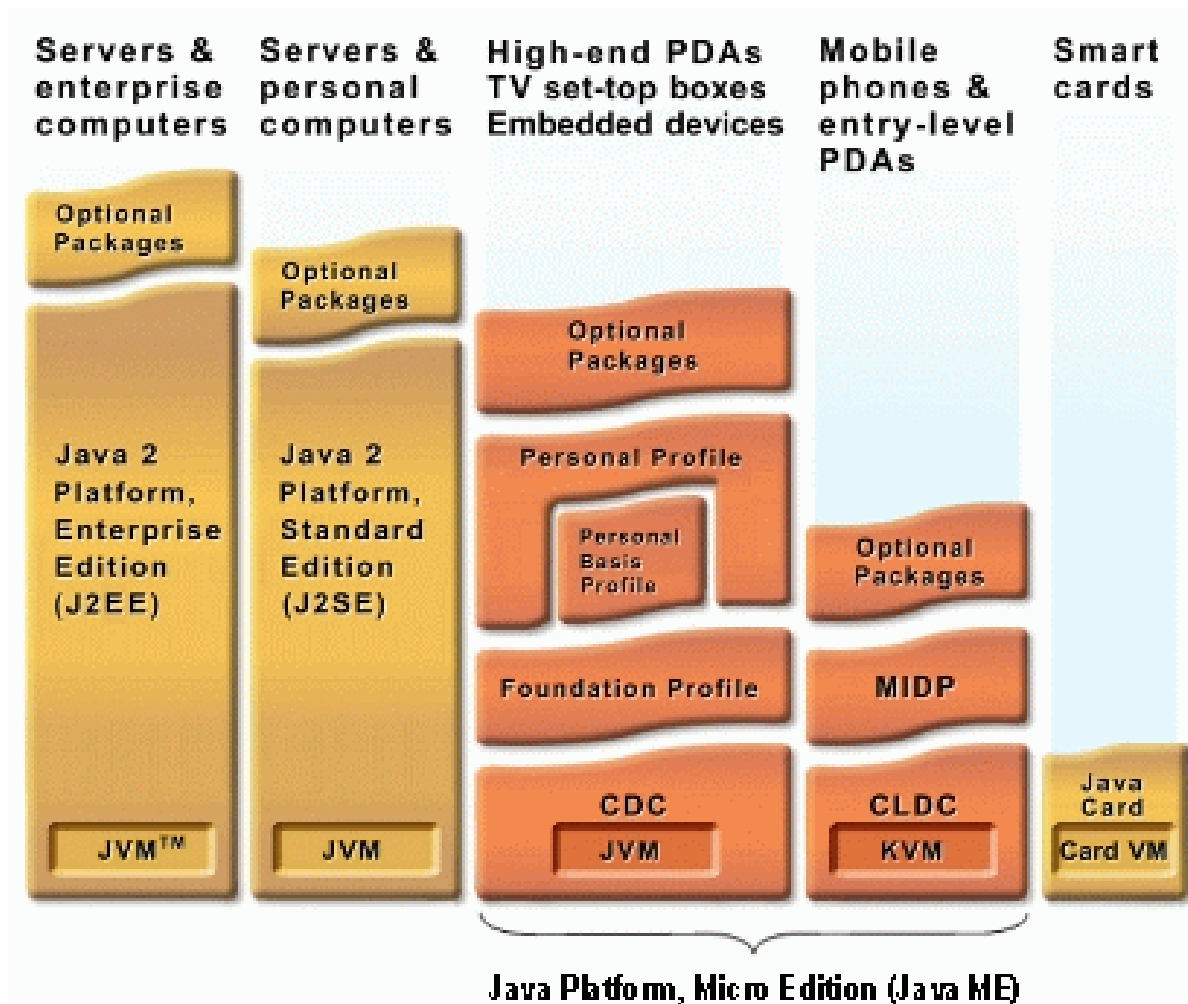
Wikipedia: Java virtual machine (sep 2014)



Wikipedia: Java virtual machine (sep 2014)



Hay (al menos) tres “grados” de Máquinas Virtuales Java



... luego esta la de Android (originalmente Dalvik, ahora ART)