Java Technology, from Netscape to Android

OUR SERVERS ARE USING TOO MUCH ELECTRICITY. WE NEED TO VIRTUALIZE.
scottadams@aol.com
www.dilbert.com
I DID MY PART BY READING ABOUT VIRTUALIZATION IN A TRADE JOURNAL. NOW YOU DO THE SOFTWARE PART.
2-12-08 © 2008 Scott Adams, Inc./Dist. by UFS, Inc.
WHY IS YOUR PART TAKING SO LONG?

Outline

```
while (today_talk) {
        The World in the ’90();
        Computer Science in ’90 ();
        The dawn of java ();
                Java technologies ();
                Introduction ();
                Memory management();
                Virtual Machines ();
                Compilation Strategies ();
        The rise of android ();
                Android implementation of VMs ();
    if (end) {
        Conclusions ();
     }
}
```

The World in the '90

- George H. W. Bush on times

- The destruction of the Berlin Wall by East Germany

- Hubble Space Telescope launched
(Space Shuttle Discovery mission)

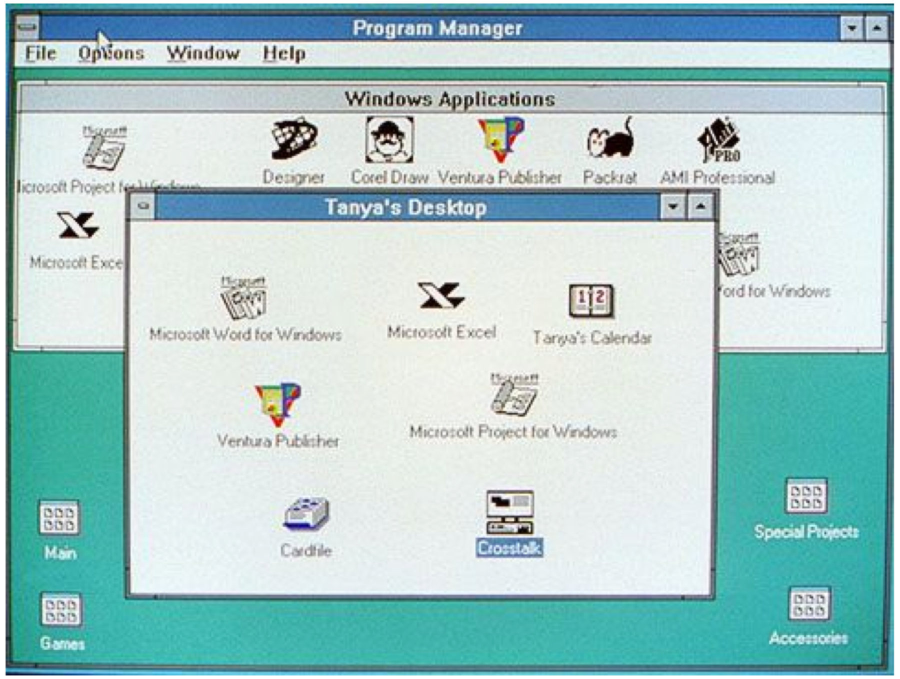
- @Cinema Back to the future

- Nintendo Game Boy on market

Computer Science in ‘90

The Linux kernel (Linus Torvalds) is released to several Usenet newsgroups.

Almost immediately, enthusiasts began developing and improving it, such as adding support for peripherals and improving its stability.

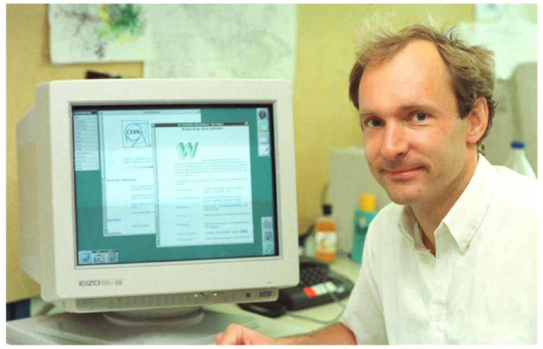


Microsoft ships the first successful version of Windows 3.0 (DOS compatible) on IBM PCs

- Better performance
- Improved GUI
- Multitasking (80386)
- Office (word, excel)

The Antefact

1990 HTML@CERN by Time Berneers-Lee
- Hypertext Markup Language



1990 PERL@UNISYS by Larry Wall
- Scripting Language

1991 – JAVA@SUN By James Gosling
- Write Once, Run Anywhere (WORA)
- Platform independent object oriented programming language

The dawn of Java 1/3

Java is used for developing all types of cross platform applications (Mobile, Desktop, server-side and dynamic web applications)

James Gosling got the idea for the first Java Machine (JV) while writing a program to port software from a PERQ by translating Perq Q-Code to VAX assembler and emulating the hardware.

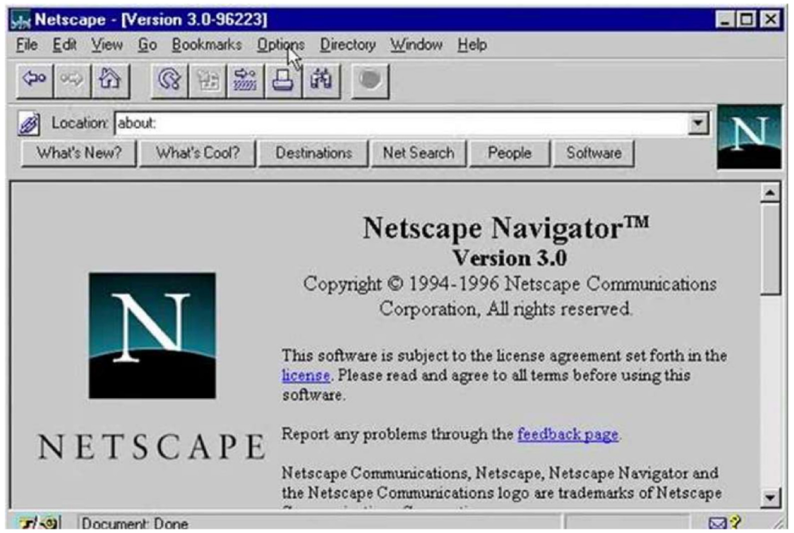
This concept was just right for the Internet, which was just starting to take off.

In 1995, Gosling team announced that the Netscape Navigator browser would incorporate Java technology.

Today, Java not only permeates the Internet, but also is the invisible force behind many of the applications and devices that power our day-to-day lives. From mobile phones to handheld devices, games and navigation systems to e-business solutions, Java is everywhere!

NetScape Navigator

Flagship product of the Netscape Communication Corp. that dominated web browser in terms of usage share between 1990-2000


Netscape - [Version 3.0-96223]
File Edit View Go Bookmarks Options Directory Window Help
Location: about:
What's New? What's Cool? Destinations Net Search People Software
NETSCAPE
Document: Done

Netscape Navigator™
Version 3.0
Copyright © 1994-1996 Netscape Communications Corporation, All rights reserved.

This software is subject to the license agreement set forth in the license. Please read and agree to all terms before using this software.

Report any problems through the feedback page.

Netscape Communications, Netscape, Netscape Navigator and the Netscape Communications logo are trademarks of Netscape

Netscape's contributions to the web include

- Javascript (which was submitted as a new standard to ECMA International
- Java support (“Netscape Plugin Application Programming Interface” NPAPI)

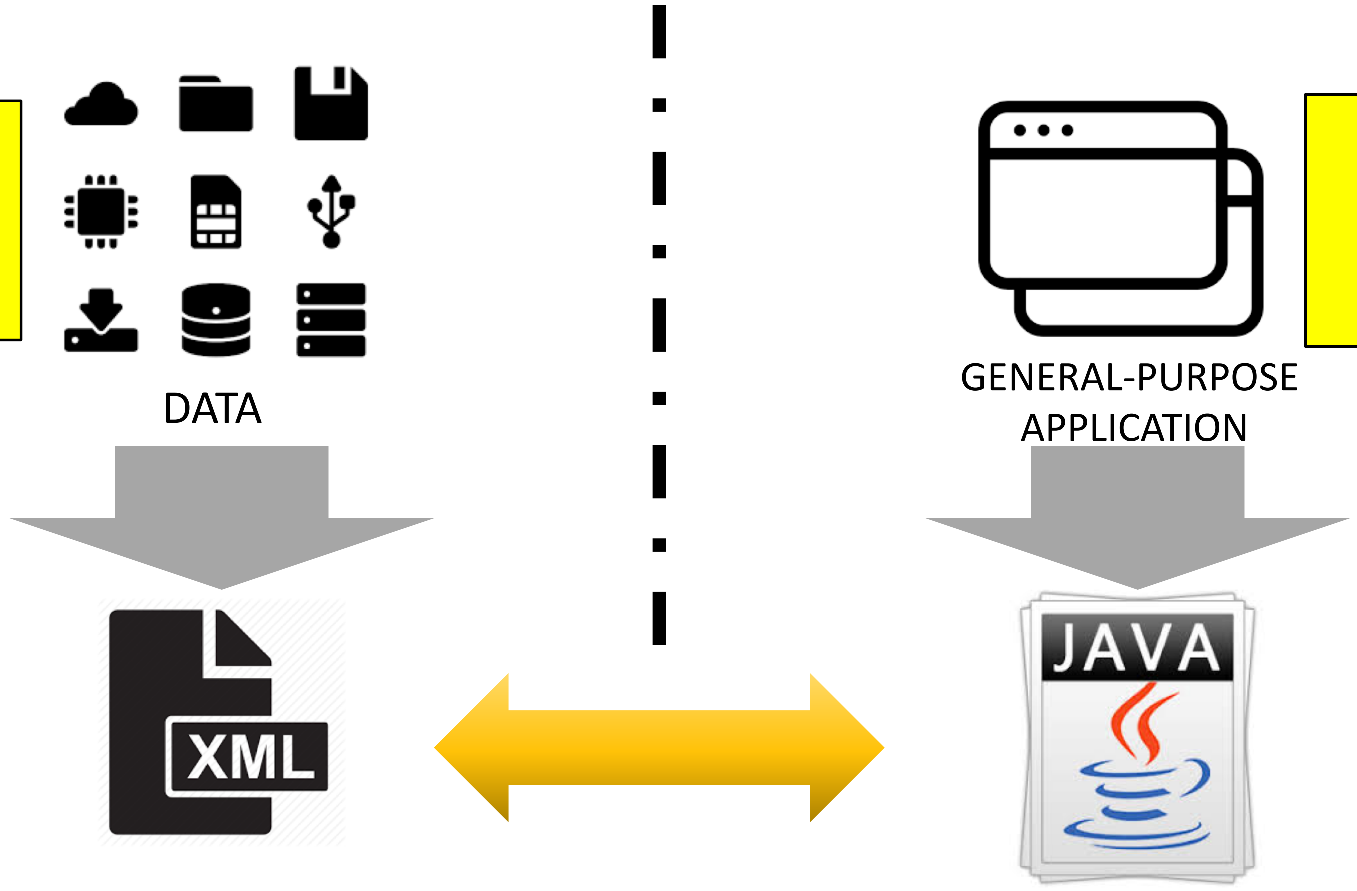
From the Plugin-store to the app store

- The rise of web usage on mobile device browsers (without plugins support), increasingly led browser makers to restrict and remove plugin support from their products to unify the set of features available across desktop and mobile versions.
- The “app store” model grew for reasons related to simplicity, security, and centralized.

The dawn of Java 2/3

XML does for data what Java does for application

XML describes data



Java describes the behavior behind the data

- XML gives Java something to do, Java lets XML do something useful
- XML/Java have reciprocal duties in enabling the Web, Java is the brain of the Internet, XML the voice

The dawn of Java 3/3

Java is not compiled to the binary instructions code recognized by a processor.

Java is translated into a Java class file which contains a stream of byte code, processed at run time by an intermediate program called Java Virtual Machine (programs, chips)

Write Once, Run Anywhere (as long you have the JVM and Java API)

From laptops to datacenters, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

Some (impressive) facts and figures:

- 97% of Enterprise Desktops Run Java
- 89% of Desktops (U.S.) Run Java
- 9 Million Java Developers Worldwide
- 3 Billion Mobile Phones Run Java
- 100% of Blu-ray Disc Players Ship with Java
- 5 Billion Java Cards in Use
- 125 million TV devices run Java

Java Technologies

These platforms are specifications; they are norms, not software!

JAVA Standard Edition (SE)

Applet, awt, Rmi, jdbc, Swing, collections, xml binding, JavaFX, Java streaming

Development tools, deployment technologies, and other class libraries and toolkits used in Java applications

JAVA Enterprise Edition (EE)

Servlet, websocket, java faces, dependency injection, ejb, Persistence, Transaction, Jms, batch api

Built on top of the Java SE platform. The Java EE platform provides an API and runtime environment for developing and running large-scale, multi-tiered, scalable, reliable, and secure network applications.

JAVA Micro Edition (ME)

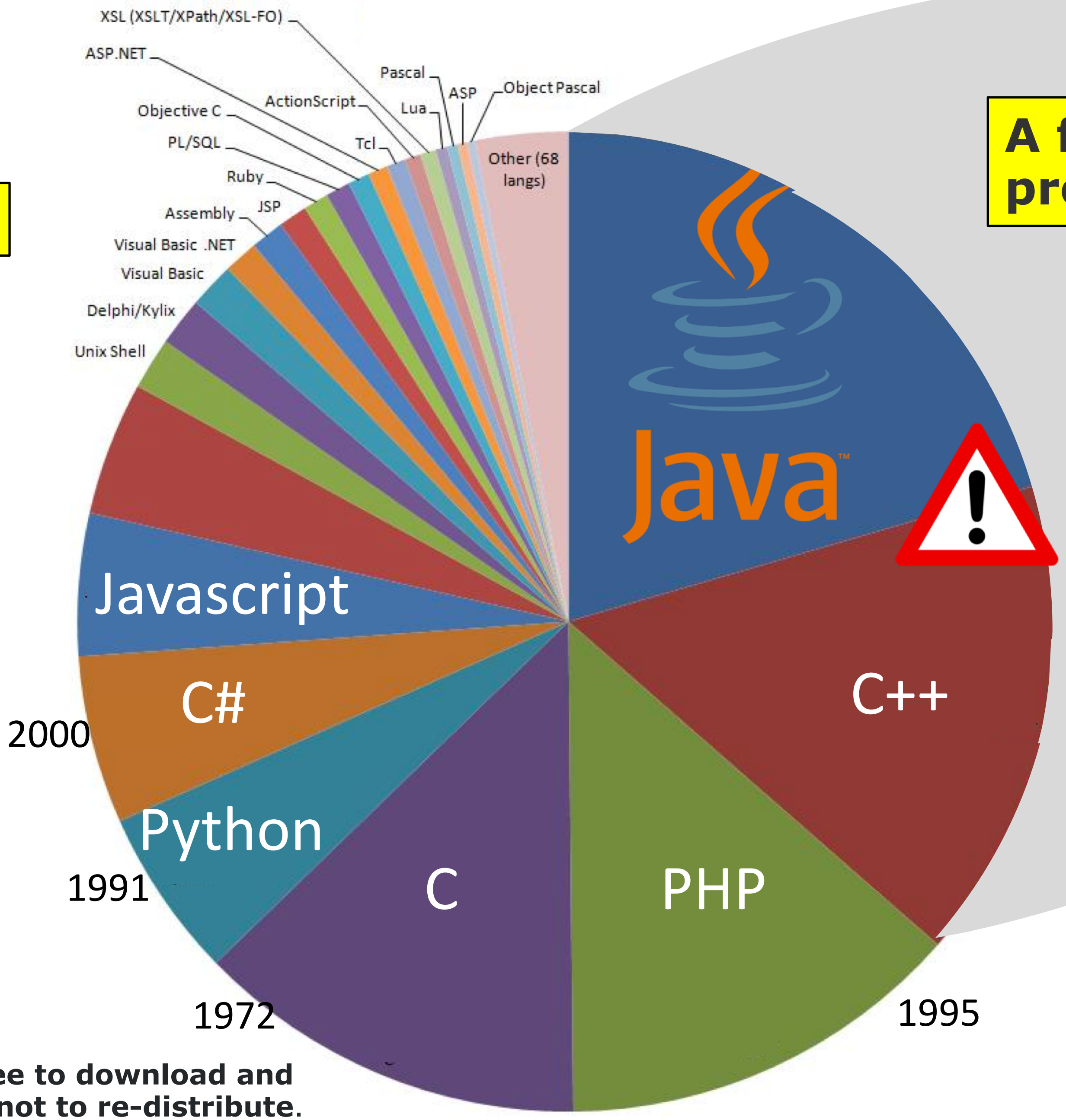
Wireless Messaging, Java ME Web Services, Security and Trust Services API Location, Mobile XML API

The Java ME platform provides an API and a small-footprint virtual machine for running Java programming language applications on small devices, like mobile phones. The API is a subset of the Java SE API, along with special class libraries useful for small device application development. Java ME applications are often clients of Java EE platform services.

Java Technologies

Java is free*

- In most scenarios you don't have to pay for almost anything except for the developers.
- There are a lot of free and open source Production Grade tools for java.

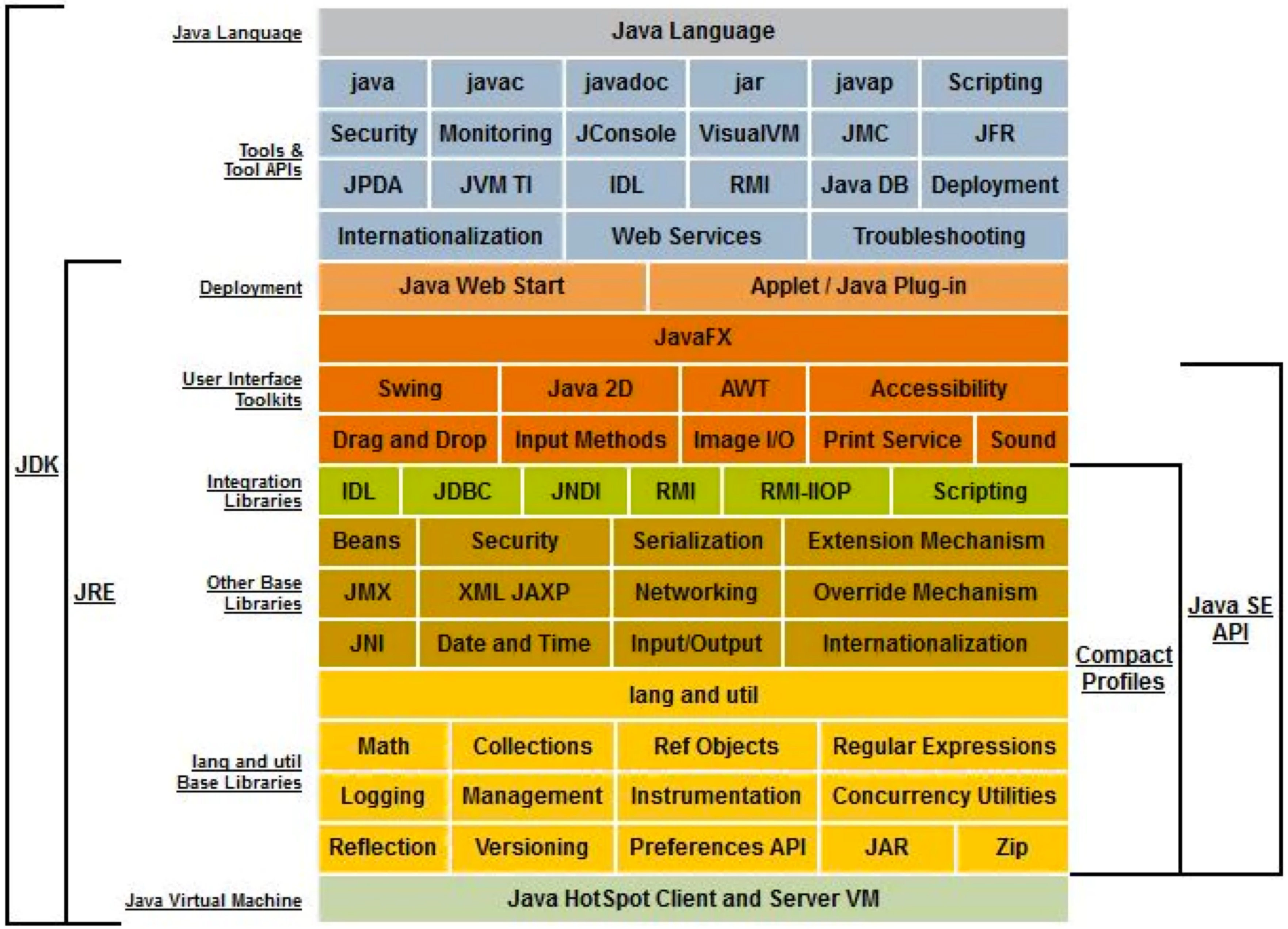


A fast and reliable programming language

Java is nearly as fast as **C++**. It has been refined over the last 20 years and is one of the most stable languages out there.

*The Java Development Kit (JDK) is **free to download and use** for commercial programming, but **not to re-distribute**.

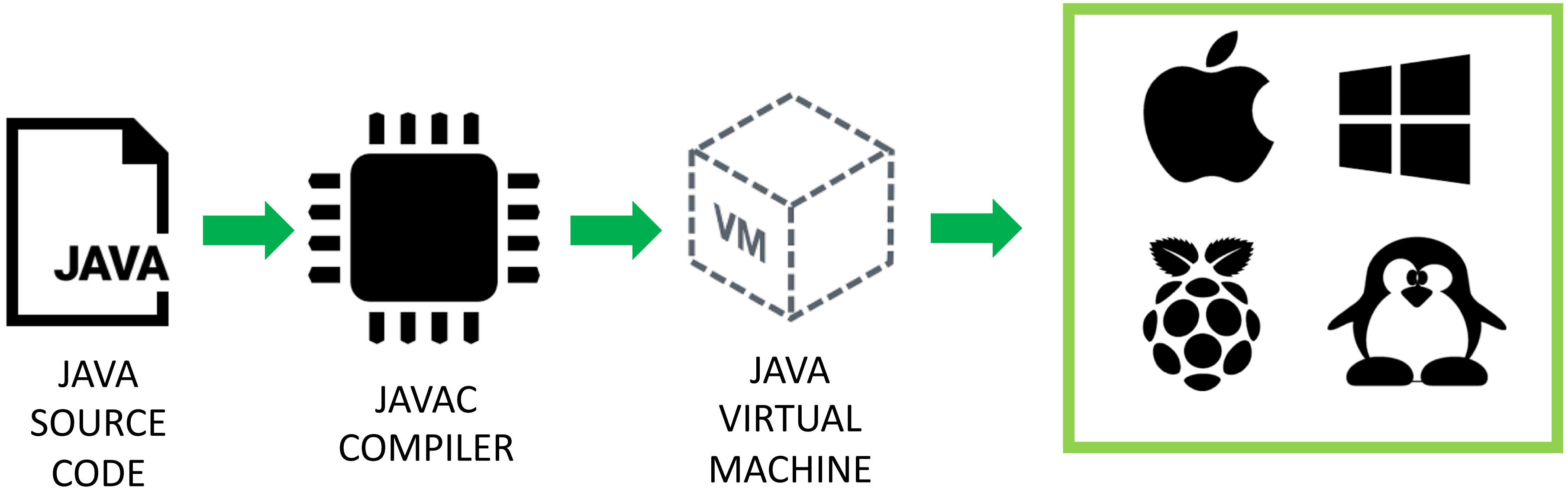
Java Technologies: the bigger picture



The Java Virtual Machine (JVM)

A VM is a high-level abstraction on top of the native operating system, that emulates a physical machine.
The JVM serves as an interpreter between Java's run-anywhere bytecode and your computer's own system.

- The JVM provides a run-time environment in which Java bytecode can be executed.
- The JVM <u>walks</u> your computer through the execution of bytecode instructions.
- The JVM <u>examines</u> your bytecode and carries out the instructions described in the bytecode.
- The JVM <u>interprets</u> bytecode and makes Java more portable than programs in any other language.

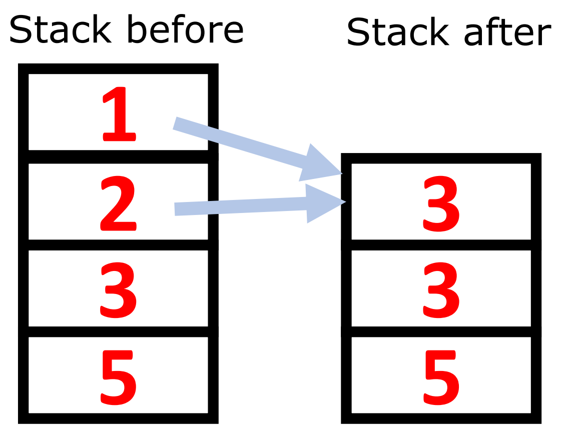


A JVM is distributed along with Java Class Library, a set of standard class libraries (in Java bytecode) that implement the Java Application Programming Interface (API).
These libraries, bundled together with the JVM, form the Java Runtime Environment (JRE)

The Java Virtual Machine(s)

Stack Based (JVM, .NET)

The memory structure where the operands are stored is a stack data structure



```
1. POP 1
2. POP 2
3. ADD 1, 2, result
4. PUSH result
```

The overhead of pushing to and popping from a stack slows down the VM

Register Based (Lua, Dalvik)

- The data structure where the operands are stored is based on the registers of the CPU.
- There is no PUSH/POP operations, but the instructions need to contain the addresses (the registers) of the operands

Register before

R1	1
R2	2
R3	3
R4	5

```
ADD R1, R2, R3
```

Register after

R1	1
R2	2
R3	3
R4	5

Faster approach, also common expression calculated can be stored for future use.
(but average register instruction > average stack instruction)

Compilation Strategies

Ahead-of-time AOT

Compile a higher-level (C,C++,Java bytecode) into a native (system-dependent) machine code -> the resulting binary file can execute natively.

Just-in-time (JIT)

Compile using the two traditional approaches to translation to machine code and interpretation.

- A JIT compiler may translate Java bytecode into native machine language while executing the program.
- The translated parts of the program can then be executed much more quickly than they could be interpreted.
- This technique gets applied to those parts of a program frequently executed.
- JIT compiler can significantly speed up the overall execution time.

Tracing Just-in-time (JIT)

Optimizing the execution of applications by continually profiling the applications each time they run and dynamically compiling frequently executed short segments of their bytecode into native machine code.

1. Identify hot loops
2. Tracing phase
3. Optimization
4. Compilation

The Rise of Android

Android is an operating system developed by Google, based on a modified version of the Linux Kernel.

- Initially developed by Andy Rubin et al. @Android Inc., later (2005) bought by Google
- It adopts the Linux Kernel 4.4 although it does not support X, GNU
- Applications are written using the Android SDK and Java (together with C/C++/Go/Kotlin)

Impressive facts and figures

- 3,617,779 app!
- 2 billion monthly active users
- 87.7% share of the global market
- 1.5 million daily Android activation
- 4000 different Android devices (500 carriers)

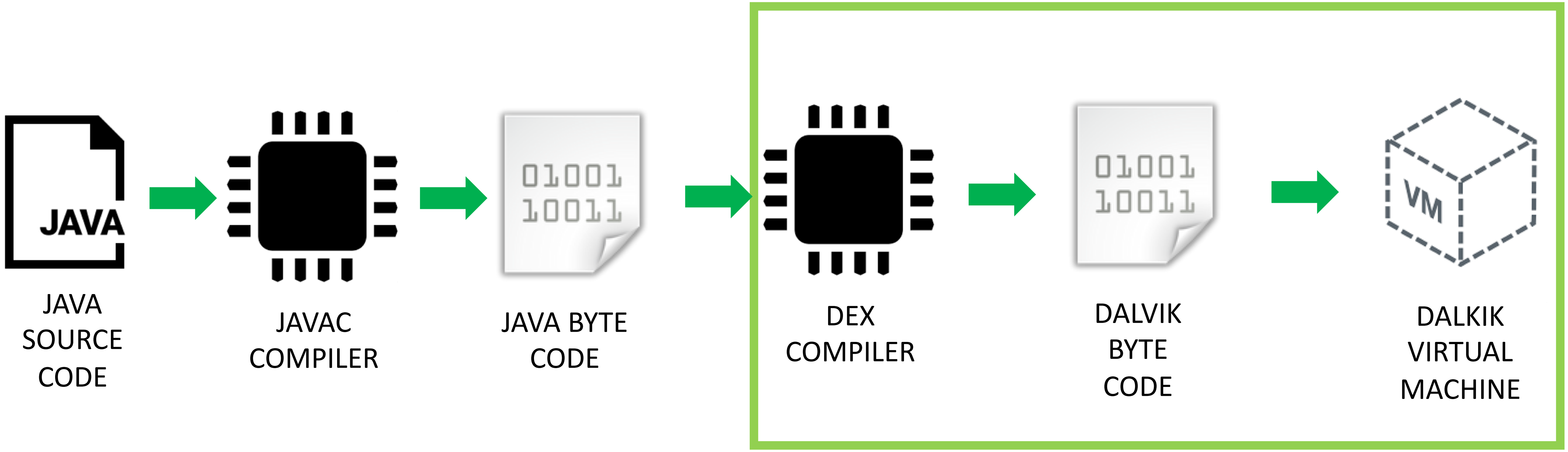
Why Java?

- Write once, run anywhere!
- A large swath of developers is the Java Community Process (JCP), which is the mechanism for developing standard technical specification for Java technology. Thanks to the JCP, the Java language has been extensively revised to address all kinds of different programming problems
- Large number of open source libraries and development tools available to make developers' lives easier.

The Android heart: Dalvik VM

Dalvik is a register-based VM that executes applications written for Android

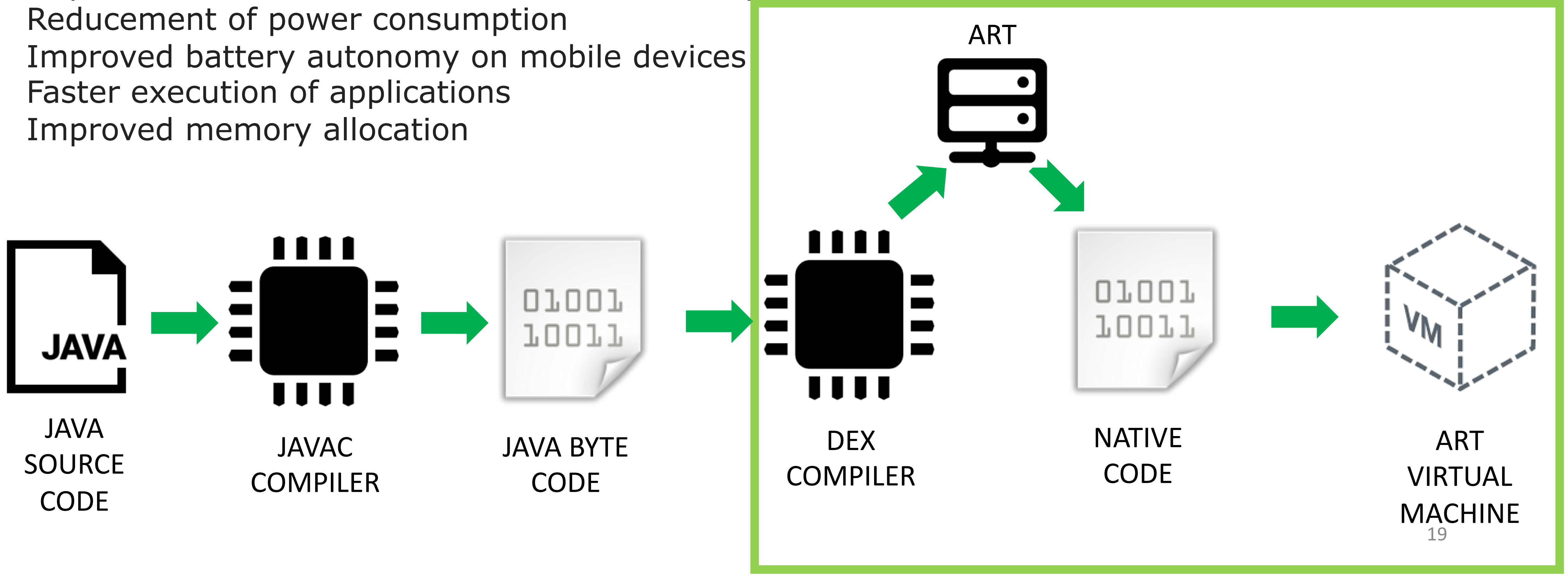
- Linux kernel of the Android OS spawns a new Dalvik VM instance for every process to improve general stability
- Since Android 2.2 Dalvik featured Tracing-JIT
- After Android 4.3 Dalvik was abandoned



The Android new heart: Android Runtime (ART) VM

ART performs the translation of the application's bytecode into native instructions (AOT), later executed by the device's runtime environment

- Increased time (for the compilation) when an application is installed,
- Increased application space of media storage
- Supports the same input bytecode as Dalvik (backcompatible)
- Improvement of the overall execution efficiency
- Reducement of power consumption
- Improved battery autonomy on mobile devices
- Faster execution of applications
- Improved memory allocation



Conclusions

- The world in 2020 is very different from ‘90 although most of modern languages are 30 years old!
- Computers hardware got (thousand) times better, software and operating system got complex and sophisticated too.
- Java technologies demonstrated to be effective in the course of the last decades
    - Java portability, scalability, modularity and open community made a hit
    - Java enabled the World Wide Web as we know it
    - The client-side java plugins (NPAPI) had a great past while now phased out
    - The new life of the defunct NPAPI plugins are the Android App, still Java based!
- Now it’s up to you to shape the future of the computing, HAPPY CODING!