@MyWebUniversity



Java

Write Once, Run Anywhere

the JVM · Android · banks · big data

JAMES GOSLING · 1995

A MYWEBUNIVERSITY STORY

Java

Write Once, Run Anywhere

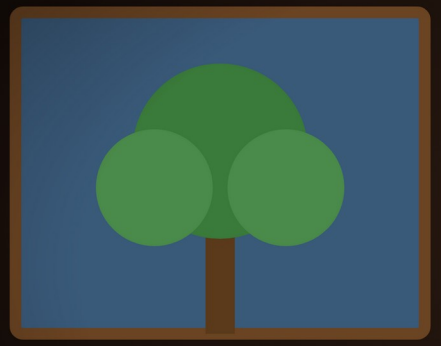
JVM · Android · enterprise · big data



James Gosling

Photo: via Wikimedia Commons

The Green Project, 1991



"Oak" — the tree outside the window

set-top box
CPU A

PDA
CPU B

phone
CPU C

different processors everywhere

Green Project · Sun Microsystems · 1991

James Gosling · Mike Sheridan · Patrick Naughton

- 1991: Green Project at Sun Microsystems
- James Gosling, Mike Sheridan, Patrick Naughton
- Many processors → need one portable language ("Oak")

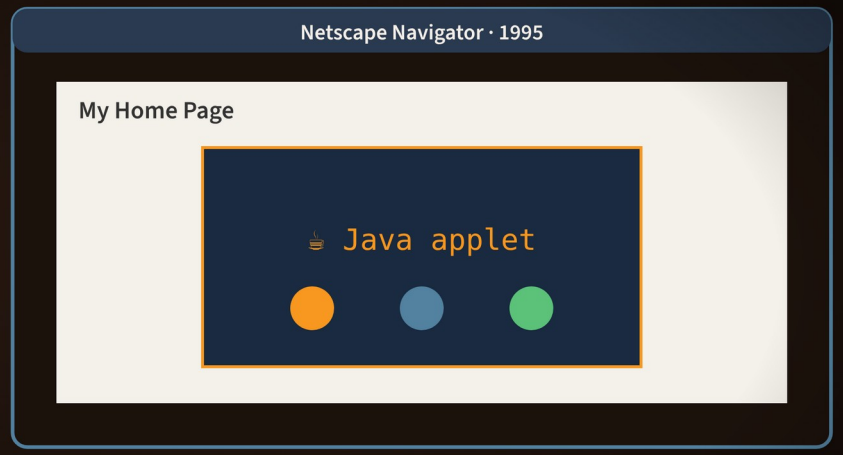
01

Oak Becomes Java, 1995

~~Oak~~

Java



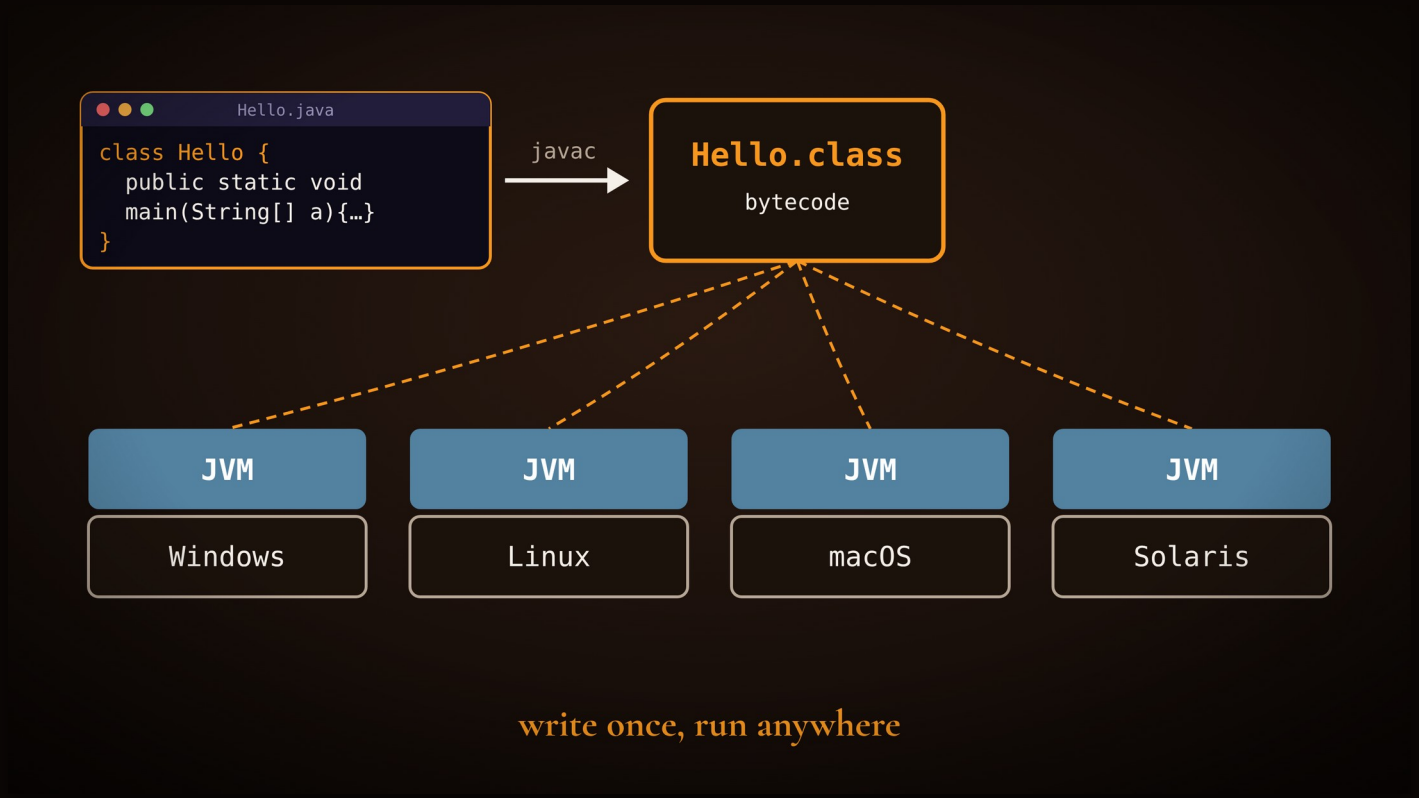


May 23, 1995 · Java announced

- "Oak" was taken → "Java" (coffee)
- May 23, 1995: Java announced · Netscape support
- Applets ran inside web pages

02

The Java Virtual Machine



- javac → bytecode (.class)
- A JVM for every OS runs the same bytecode
- Write once, run anywhere

03

Safer than C++

```c
int *p = malloc(10 * sizeof(int));
p[20] = 5;      /* oops */
free(p); free(p); /* crash */
```

C/C++: manual memory, pointer arithmetic

Safe.java

```java
int[] a = new int[10];
a[20] = 5;
// ArrayIndexOutOfBoundsException
// memory freed by the GC
```

Java: bounds checks + garbage collector



- No pointer arithmetic · garbage collection
- Classes, strong typing, bounds checks
- Big standard library: network, threads, files

04

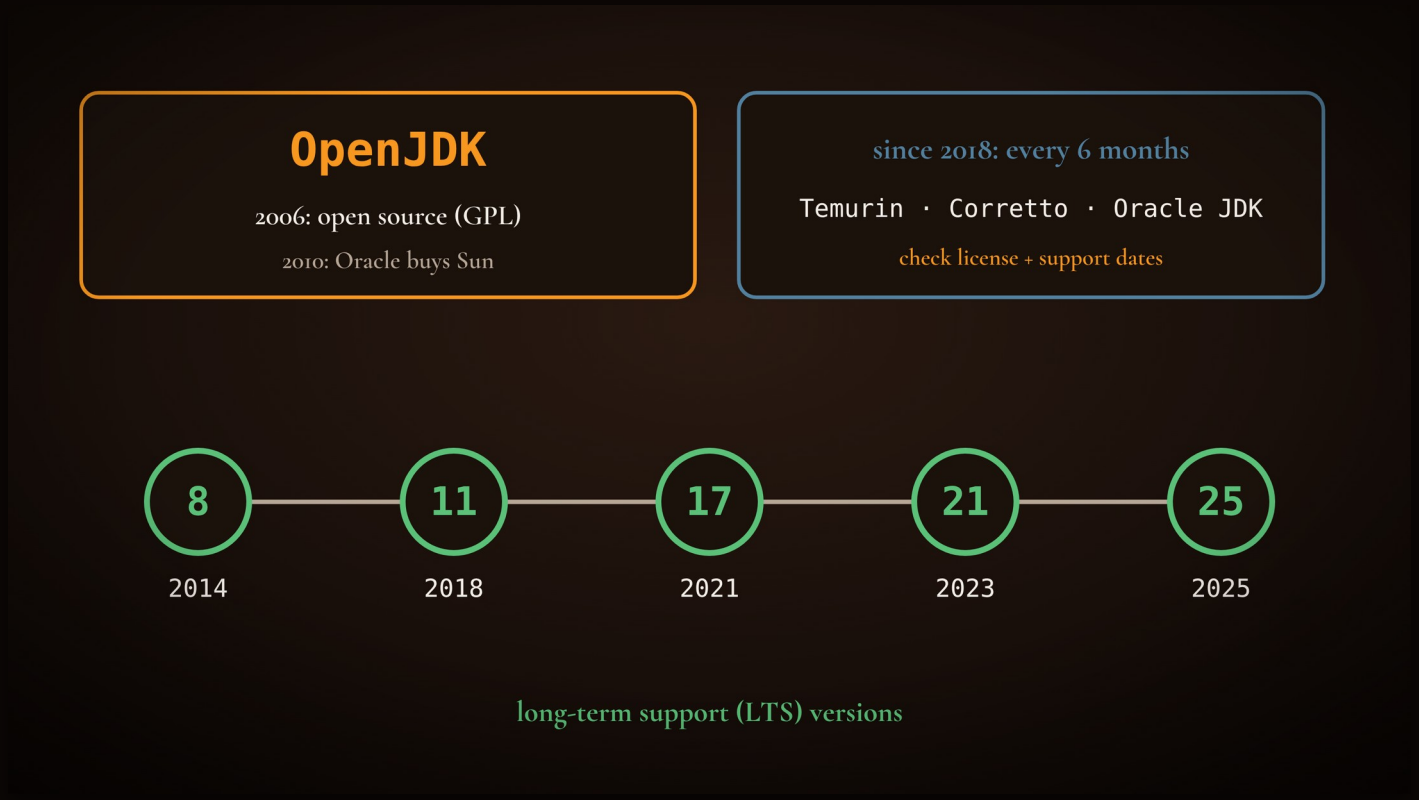
Java Runs the Enterprise



- Banking, insurance, airlines, government
- WebLogic · Tomcat · Spring · Hadoop · Kafka
- Android: Java and Kotlin

05

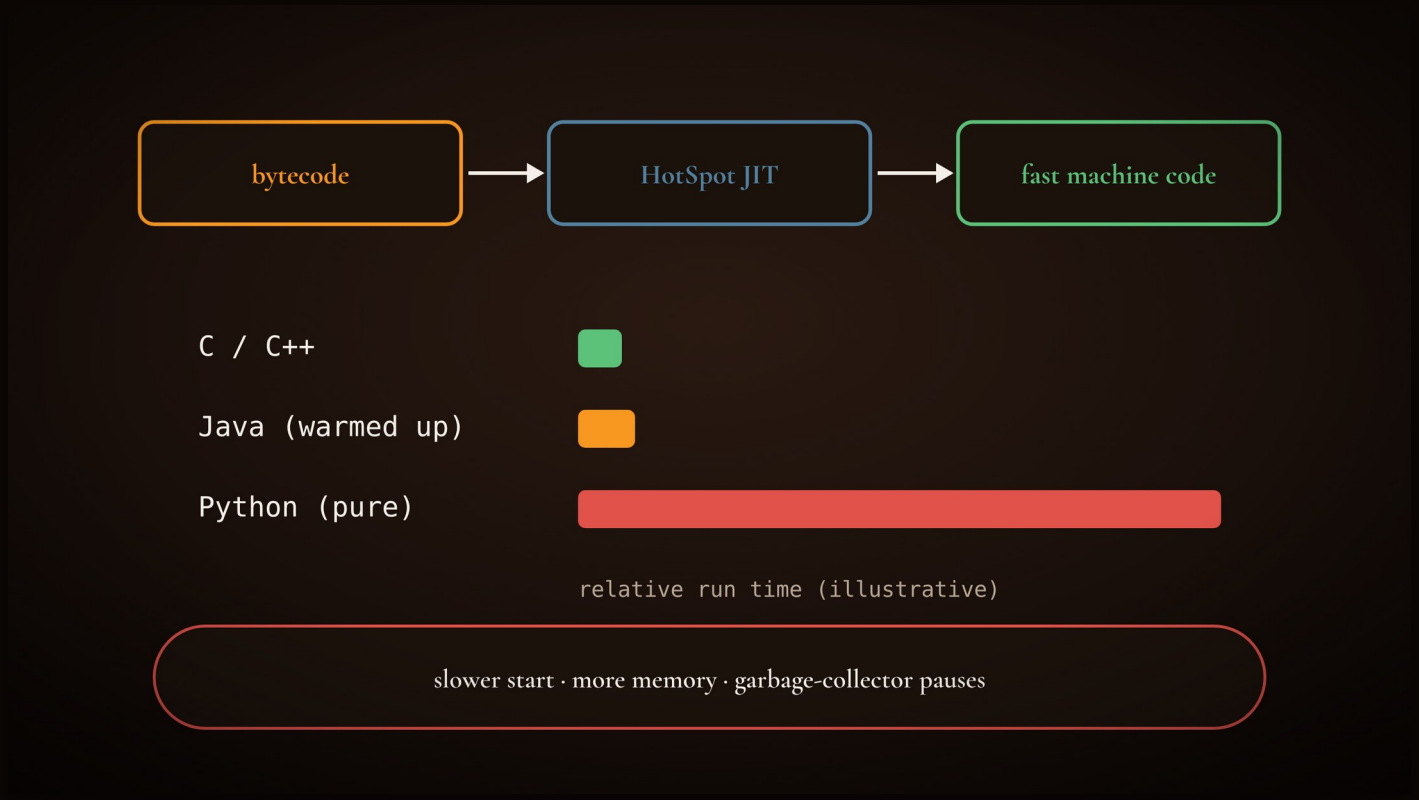
OpenJDK and Oracle



- 2006–2007: OpenJDK (GPL) · 2010: Oracle buys Sun
- Since 2018: release every 6 months
- LTS: 8 · 11 · 17 · 21 · 25 · OpenJDK builds (Temurin, Corretto)

06

How Fast Is Java?



- HotSpot JIT compiles hot code at run time
- Long-running servers: close to C++
- Slower start · more memory · GC pauses

07

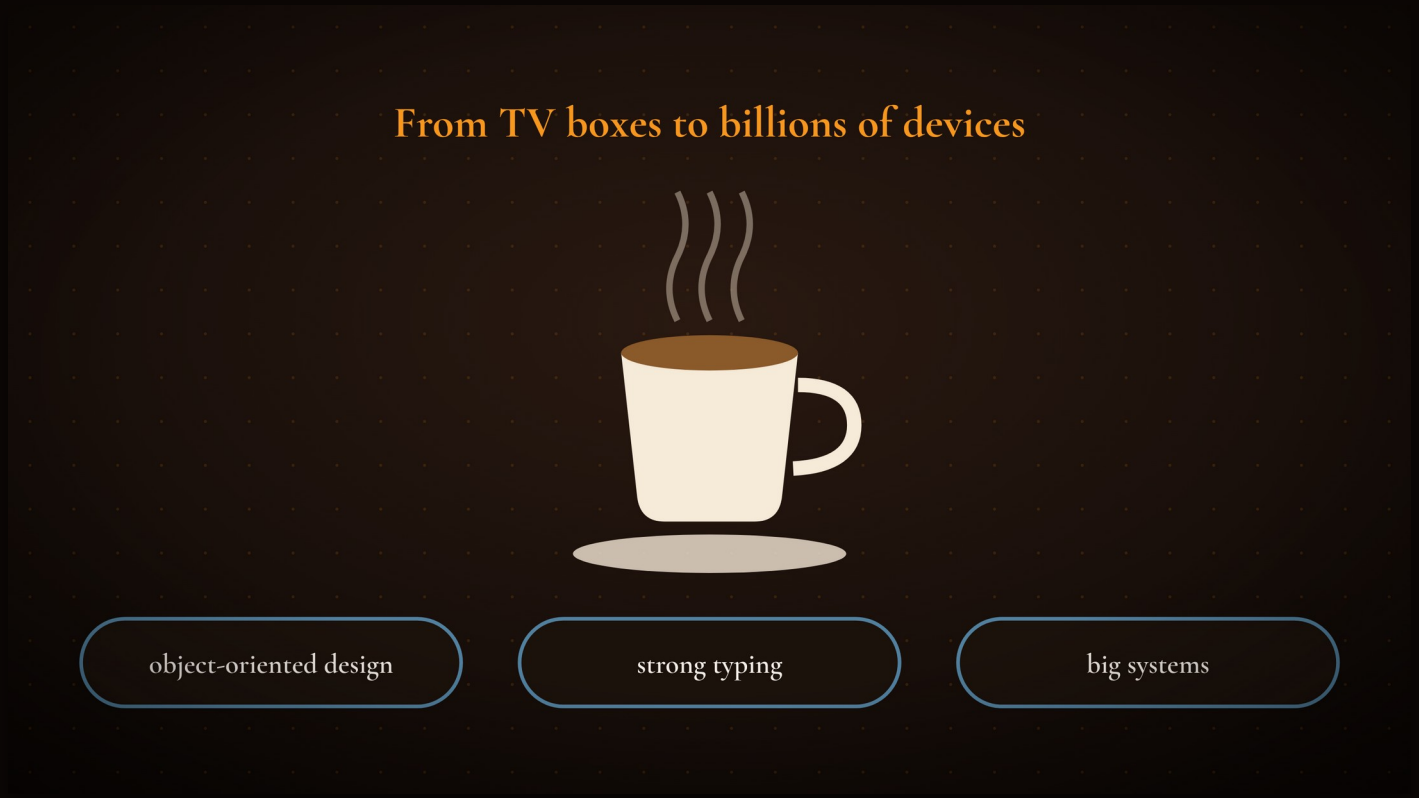
When to Choose Java



- Java: large, long-living team systems
- Python: scripts and data · C/C++/Rust: OS, drivers, games
- Keep Java and libraries patched (Log4Shell, 2021)

08

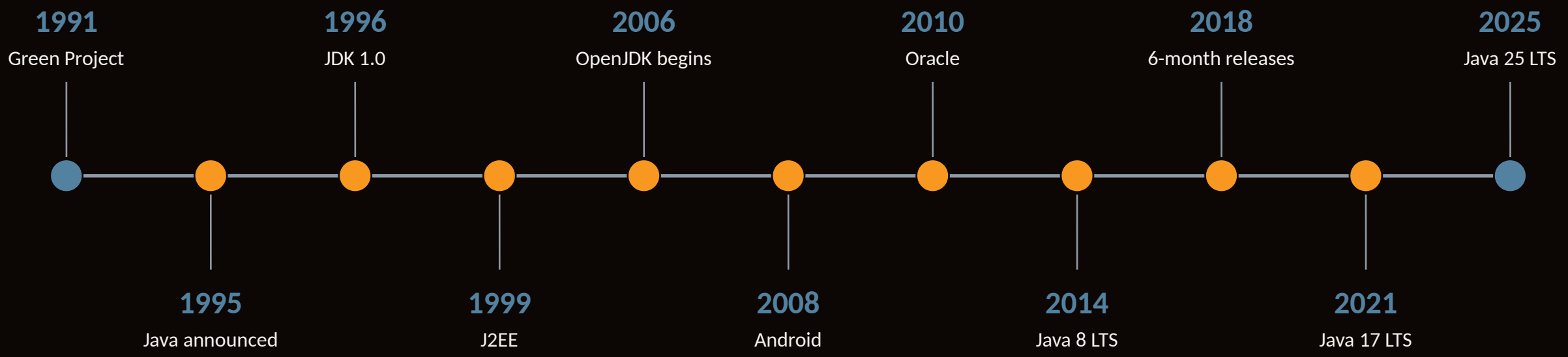
Thirty Years of Java



- From TV boxes to billions of devices
- Learn OOP, strong typing, big-system design
- Jobs in almost every industry

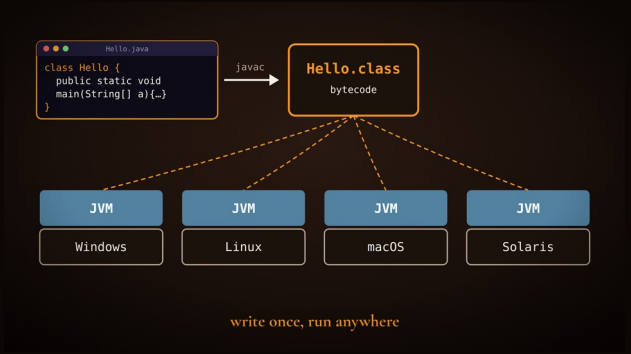
09

Timeline: 1991 to 2025



Sources and Further Reading

1. James Gosling, Java history — oracle.com/java
2. OpenJDK project — openjdk.org · JDK release cadence (JEP 322 era)
3. Oracle Java SE Support Roadmap — oracle.com/java/technologies/java-se-support-roadmap.html
4. Eclipse Adoptium (Temurin) — adoptium.net · Amazon Corretto — aws.amazon.com/corretto
5. CISA, Apache Log4j Vulnerability Guidance (2021)
6. Photo: James Gosling — via Wikimedia Commons (original author and license listed there).





Thank you for watching

MyWebUniversity.com

Free technical education since 2000 — English and Dari

@MyWebUniversity · @MyWebUniversity-Dari