Java

Overview

- ❏ Design Goals
- ❏ Java Environment
- ❏ Java Platform
- ❏ Java Virtual Machine
- ❏ Java Architectures
- ❏ JavaOS Architecture

Design Goals

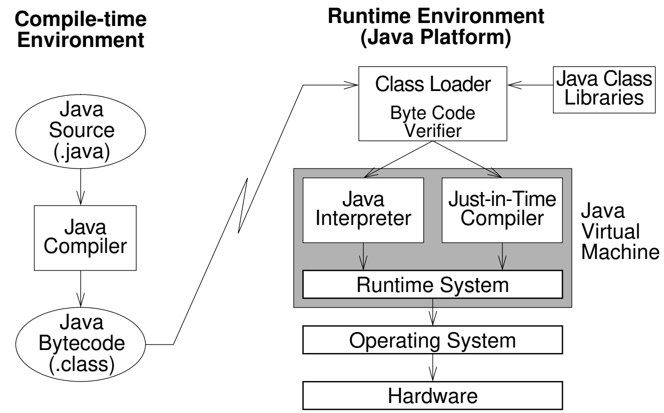
Buzzwords ...

- ❏ Simple, object oriented, and familiar
- ❏ Robust and secure
- ❏ Architecture neutral and portable
- ❏ High performance
- ❏ Interpreted, threaded, and dynamic

Why Java?

- ❏ More and more problems with development and distribution of traditional SW
  - various operating systems (Unix, Windows, MacOS, ...)
  - various hardware architectures
- ❏ Java: language and environment for *secure*, *high performance*, and highly *robust* applications on *multiple platforms* in *heterogenous, distributed networks*

Java Environment



Java Platform

Class Libraries = Java Packages

➥ **Application Programming Interfaces**

- ❏ **Java Core API**
  - Minimal set of API that form the standard Java platform
- ❏ **Java Standard Extension API**
  - Additional interfaces outside the Core API for which JavaSoft has defined and published an API standard
- ❏ **PersonalJava API**
  - for network-connectable applications on personal consumer devices (hand-held computers, set-top boxes, game consoles, smart phones, ...)
- ❏ **EmbeddedJava API**
  - for high-volume embedded devices (process control, instrumentation, ...)

Java Virtual Machine

Interpreter

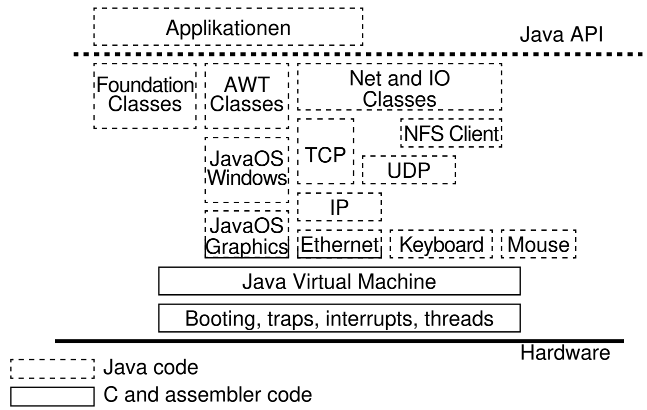
- Bytecode interpreter
- Security and safety checks

Run-time system

- Mapping byte codes to more complex functions
- Garbage collection
- *Native-code* extensions to the virtual machine
  - Examples: network and file-system interface, Motif interface

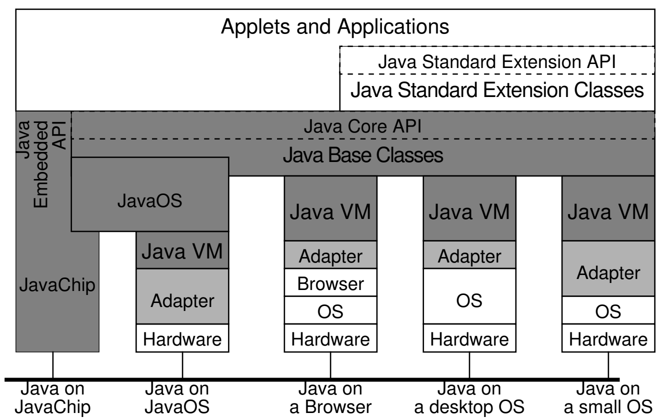
5-Java.doc: 05.09.97 12:31

JavaOS Architecture



5-Java.doc: 05.09.97 12:31

Java Architectures — Overview



5-Java.doc: 05.09.97 12:31