

Bahasa Pemrograman 2

Introduction

anton@ukdw.ac.id

Deskripsi

- Mata kuliah ini membahas tentang dasar-dasar bahasa pemrograman yang berorientasi obyek dimulai dari prinsip-prinsip class, obyek, inheritance, polymorfisme, abstraksi, interface, collections, error handling sampai dengan pengolahan file JAR menggunakan bahasa pemrograman **Java**

Keterangan

- SKS : 3
- Dosen: Antonius Rachmat C, S.Kom, M.Cs
- Email: anton@ukdw.ac.id
- YM: antonie_oo
- Web: <http://lecturer.ukdw.ac.id/anton>
- Hari Kuliah: setiap hari pukul 08.00 – 10.00

Kompetensi

- Mahasiswa dapat menjelaskan konsep pemrograman berorientasi obyek
- Mahasiswa dapat menggunakan tool bahasa pemrograman OOP -> bahasa pemrograman Java
- Mahasiswa dapat membuat program GUI dengan menggunakan Java / Netbeans
- Mahasiswa dapat merancang dan menganalisis masalah secara berorientasi obyek
- Mahasiswa dapat menggunakan class diagram

Silabus

- 1. Overview dan Pengenalan Java 1
- 2. Pengenalan Java 2
- 3. Class dan Obyek
- 4. Mendesain Class & Obyek
- 5. Inheritance
- 6. Interface & Abstract
- 7. Polymorphism
- 8. Relasi Antar Class
- 9. Collections
- 10. Exception Handling
- 11. Class Diagram
- 12. JAR File

Acuan

- Weisfeld Matt, The Object Oriented Thought Process 2nd Edition, Sams Publishing, 2004
- Poo, Kiong, Ashok Swarnalatha, Object Oriented Programming and Java 2ndEdition, Springer, 2007
- Keogh, Giannini Mario, OOP Demystified, McGraw-Hill, 2004
- Bahasa Pemrograman Java 2 by Abdul Kadir
- Diktat Kuliah Bahasa Pemrograman 2 by Katon Wijana

Bobot Penilaian

- TTS: 25%
- TAS: 30%
- Tugas-tugas Praktikum: 35%
- Tes Kecil: 10%
 - Sebelum TTS dan Sebelum TAS

Nilai

A : 85 – 100

A- : 80 – 84

B+ : 75 – 79

B : 70 – 74

B- : 65 – 69

C+ : 60 – 64

C : 55 – 59

D : 45 – 54

E : 0 – 44

Tools

- Jcreator
 - <http://www.jcreator.com>
- Java Software Development Kit (JDK)
 - <http://java.sun.com/javase/downloads/widget/jdk6.jsp>
- Netbeans 6.8
 - <http://netbeans.org/downloads/index.html>
- Eclipse
 - www.eclipse.org/downloads
- UML Editor
 - www.staruml.sourceforge.net

Peraturan

- Keterlambatan maksimal **15 menit**
- Harap tenang selama kuliah berlangsung
- Tugas/Pekerjaan yang dicurigai hasil mencontek (sama/mirip) nilainya dibagi rata
- Tidak ada Extra Project / Tugas Tambahan
- Tidak ada TTS/TAS susulan

Generasi Bahasa Pemrograman

- Generasi 1: Machine / Binary
- Generasi 2: Assembly
- Generasi 3: (High Level)
 - Java, C++, Pascal, C
- Generasi 4 (Special Purpose Language)
 - Report Generator: Crystal Reports, RAVE
 - Forms Generator: Delphi, VB, C#, FoxPro
 - CASE: Rational Rose, Poseidon
 - DBMS: FoxPro, Oracle, SQL

Perkembangan Bahasa Pemrograman

Bahasa Mesin

- Level terendah
- Isi:
 - kode-kode mesin yg hanya dapat diinterpretasikan langsung oleh mesin komputer
- Berupa kode numerik 0 dan 1
- Microcode:
 - sekumpulan instruksi dalam bahasa mesin
- (+) : Eksekusi cepat
- (-) : Sulit dipelajari manusia

Perkembangan Bahasa Pemrograman

Bahasa Assembly

- Bahasa simbol dari bahasa mesin
- Contoh: ADD, MUL, SUB, dll
- Macro instruksi:
 - sekumpulan kode dalam bahasa assembly
- (+) : Eksekusi cepat, masih dapat dipelajari daripada bahasa mesin, file kecil
- (-) : Tetap sulit dipelajari, program sangat panjang

Perkembangan Bahasa Pemrograman

Bahasa Tingkat Tinggi

- The 3rd Generation Programming Language
- Lebih dekat dengan bahasa manusia
- Memberi banyak fasilitas kemudahan dalam pembuatan program, mis.: variabel, tipe data, konstanta, struktur kontrol, loop, fungsi, prosedur, dll.
- Contoh: Pascal, Basic, C++, Java
- (+) : Mudah dipelajari, mendekati permasalahan yang akan dipecahkan, kode program pendek
- (-) : Eksekusi lambat

Perkembangan Bahasa Pemrograman

Specific Problem Oriented

- The 4th Generation Programming Language
- Digunakan langsung untuk memecahkan suatu masalah tertentu
 - Report Generator: Crystal Reports, RAVE
 - Forms Generator: Delphi, VB, C#, FoxPro
 - CASE: Rational Rose, Poseidon
 - DBMS: FoxPro, Oracle, SQL
- Contoh: SQL untuk database, Visual Basic, Delphi

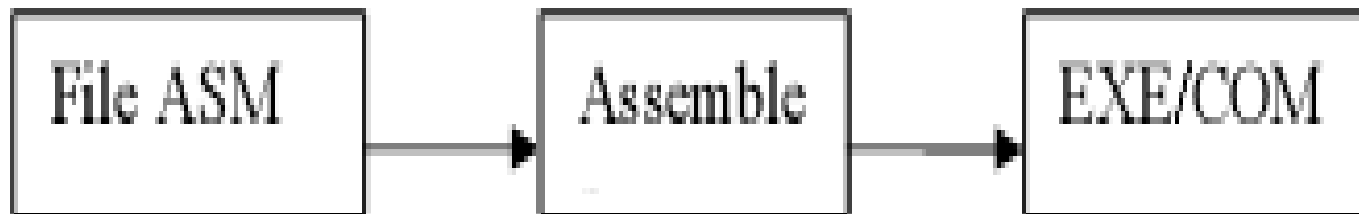
Paradigma Pemrograman

- Pemrograman Terstruktur
 - Berdasarkan urutan-urutan, sekuensial
 - Program adalah suatu rangkaian prosedur untuk memanipulasi data. Prosedur merupakan kumpulan instruksi yang dikerjakan secara berurutan.
 - Pascal, C
- Pemrograman Berorientasi Fungsi
 - Berdasarkan teori fungsi matematika
 - MathLab, F#
- Pemrograman Berorientasi Obyek
 - Java, VB.NET, C#, Object Pascal
- Pemrograman Berorientasi Komponen
 - Semua bahasa yang mendukung OOP

Macam Translator

Assembler

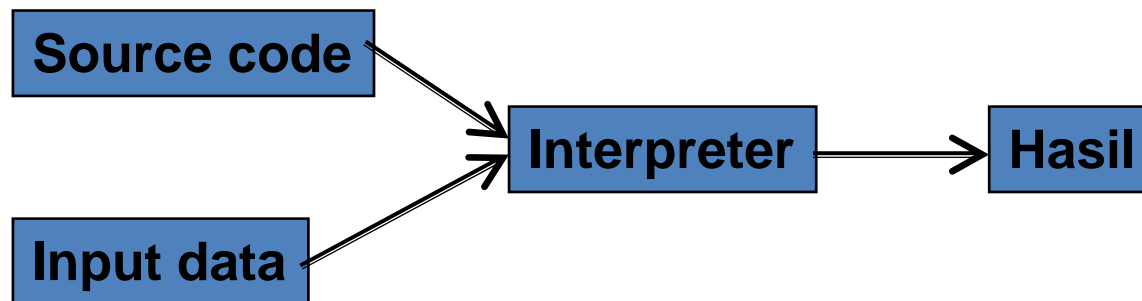
- Source code adalah bahasa assembly
- Object code adalah bahasa mesin



Macam Translator

Interpreter

- **Input**
 - *source code* : bahasa scripting (PHP, ASP, Basic, dll)
 - masukan program dari *user*
- **Output**
 - Tidak ada *object code*
 - Translasi internal



Interpreter (2)

- Program tidak harus dianalisis seluruhnya dulu tapi bersamaan dengan jalannya program
- (+) :
 - mudah bagi *user*
 - *debugging* cepat
- (-) :
 - eksekusi program lambat
 - tidak langsung menjadi program *executable*

- **Input**

- *source code* : bahasa Pascal, C, C++

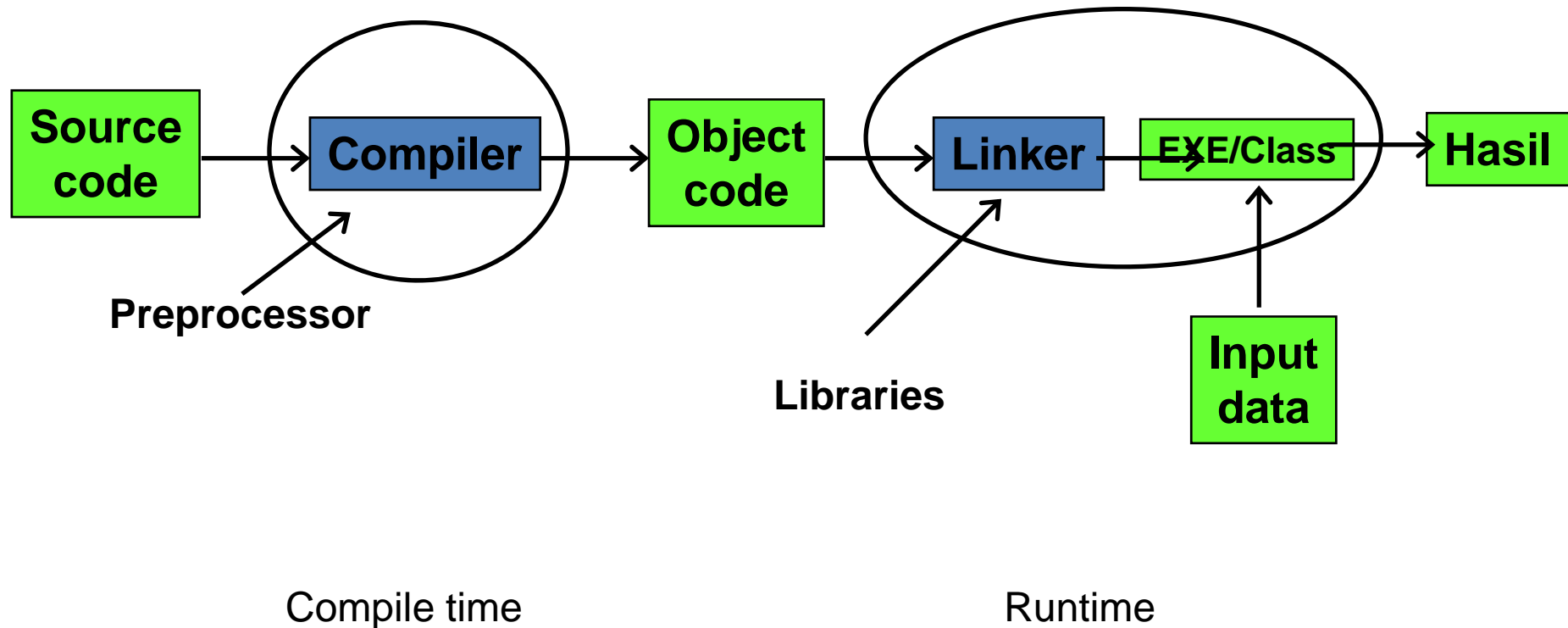
- **Output**

- *object code* : bahasa assembly atau EXE

- Compile time
 - saat pengubahan *source code* menjadi *object code*
- Runtime
 - saat eksekusi *object code*, (dan menerima *input* dari *user*)

Translator

Kompiler (3)



Contoh Kasus

- Ada 2 gelas kosong berukuran: 5 liter dan 3 liter
- Bagaimana cara kita mendapatkan air berukuran 4 liter?
- Bagaimana cara mendapatkan air berukuran 2 liter?

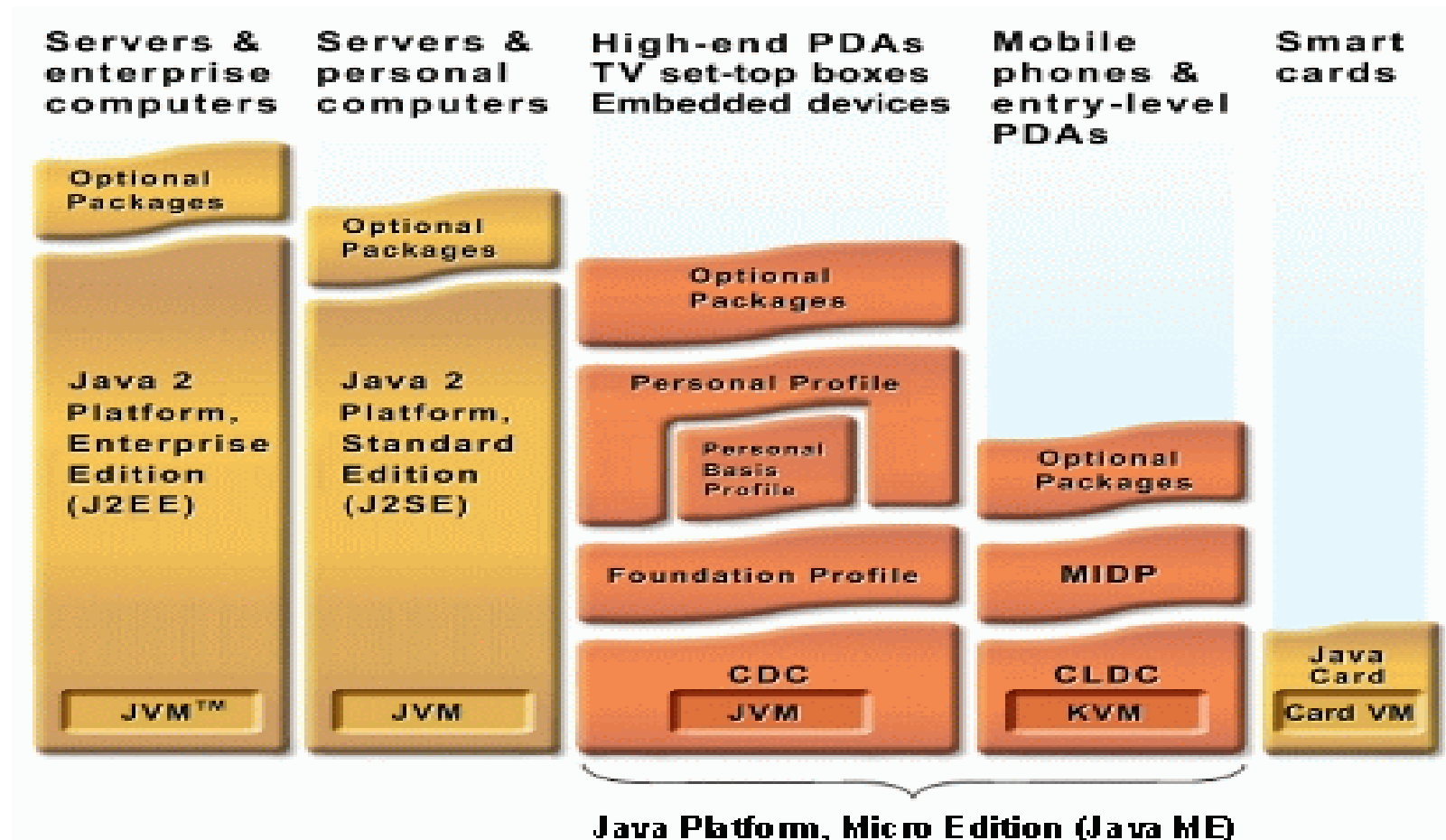
Cara

- Masukkan air ke 3 liter hingga penuh
- Masukkan air 3 liter ke 5 liter, sisa 2 liter kosong
- Masukkan air ke 3 liter hingga penuh
- Tuangkan air 3 liter ke sisa 2 liter, berarti sisa 1 liter di gelas 2 liter
- Buang seluruh air di gelas 5 liter tadi
- Tuangkan air 1 liter yg ada di gelas 3 liter tadi hingga kosong
- Masukkan air ke 3 liter hingga penuh
- Tuangkan 3 liter ke gelas 5 liter yang sudah ada 1 liter tadi, hingga kita dapat 4 liter

Java

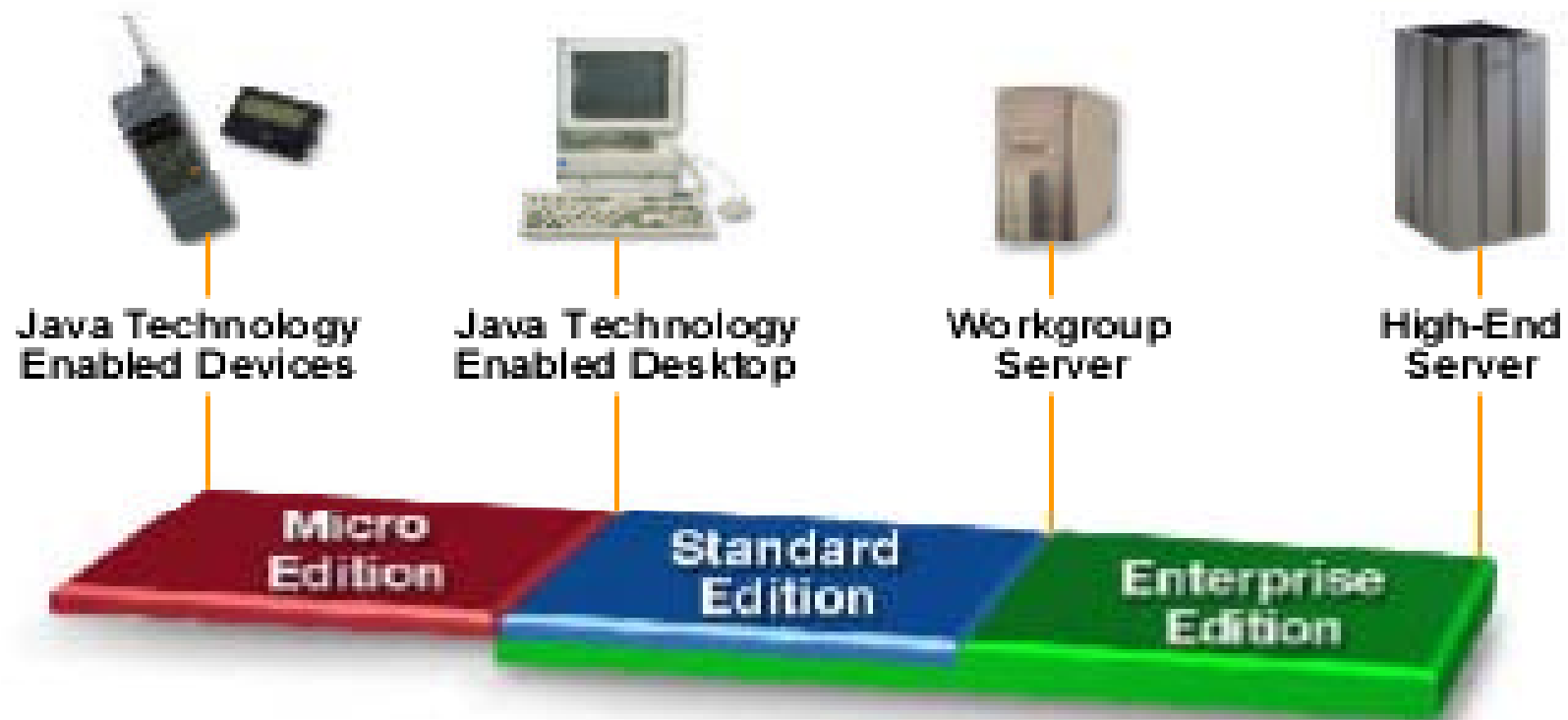
- Dibuat oleh Sun Microsystem
(<http://java.sun.com>)
- Proyek awal: Green
 - Bahasa baru: OAK oleh James Gosling
 - Kemudian oleh Sun disebut Java
- Konsep Java menggunakan OOP
 - Sifatnya: Write Once Run Everywhere?
- Mendukung multiplatform language

Jenis Java



Jenis Java

The Java™ 2 Platform



Versi Java

Java 1.02

250 classes

Slow.

Cute name and logo.
Fun to use. Lots of bugs. **Applets** are the Big Thing.

Java 1.1

500 classes

A little faster.

More capable, *friendlier*.
Becoming very **popular**.
Better GUI code.

Java 2

(versions 1.2 - 1.4)

2300 classes

Much faster.

Can (sometimes) run at native speeds. Serious, **powerful**. Comes in three flavors: Micro Edition (J2ME), Standard Edition (J2SE) and Enterprise Edition (J2EE). Becomes the *language of choice* for new enterprise (especially web-based) and mobile applications.

Java 5.0

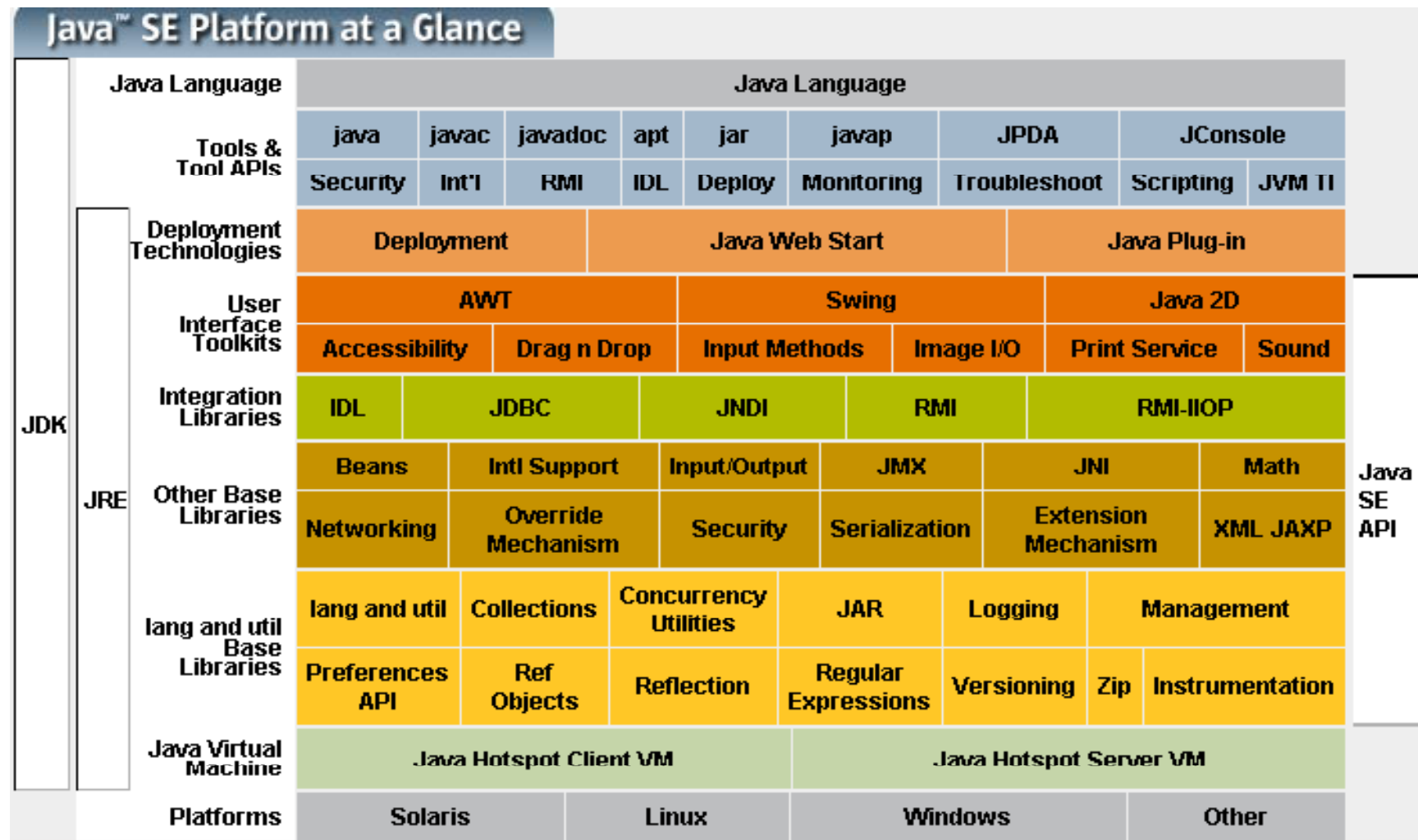
(versions 1.5 and up)

3500 classes

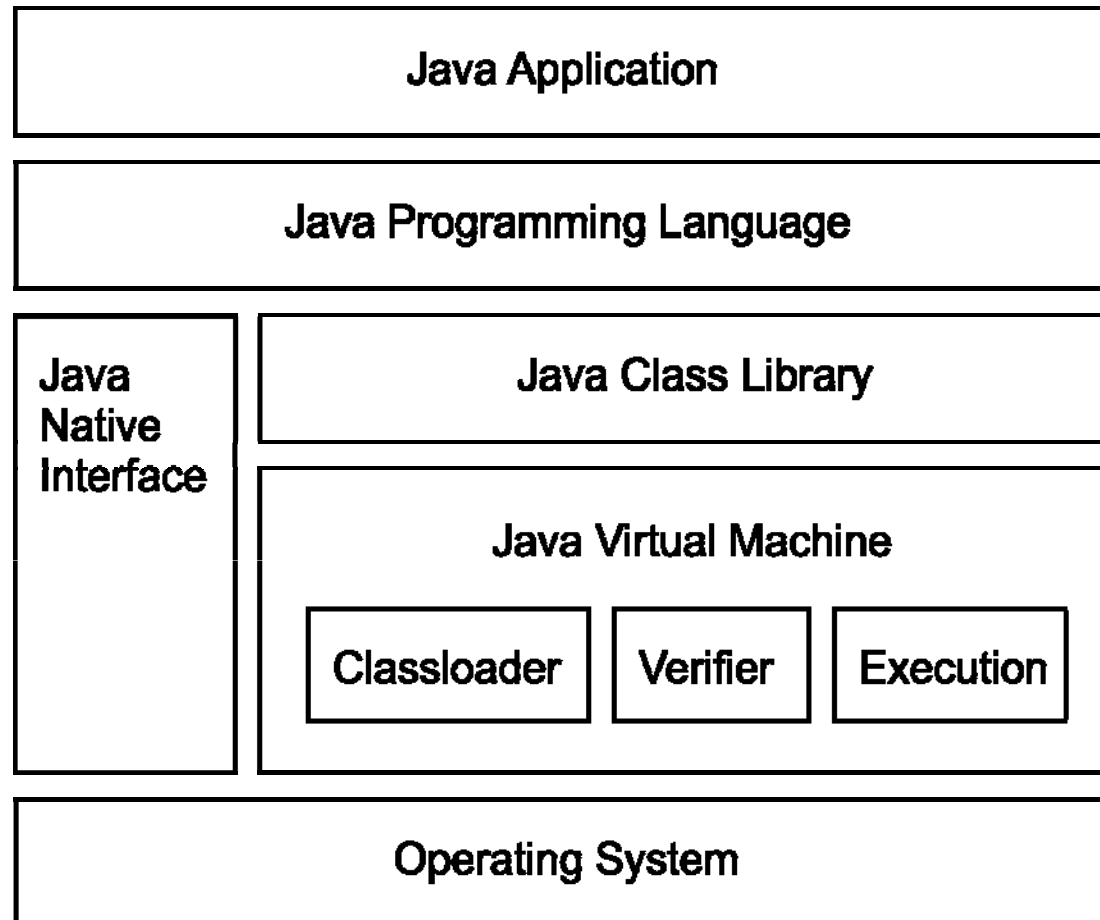
More power, easier to develop with.

Besides adding more than a thousand additional classes, Java 5.0 (known as "Tiger") added major changes to the language itself, making it easier (at least in theory) for programmers and giving it new features that were popular in other languages.

J2SE Platform



Java system overview



Yang dibutuhkan?

- J2RE untuk menjalankan Java.
 - J2RE akan berisi Java Virtual Machine.
- J2SDK untuk mendevelop Java.
 - J2SDK akan berisi JVM dan SDK
- J2ME untuk mendevelop aplikasi Mobile
- J2EE untuk mendevelop aplikasi Enterprise
 - J2EE berisi seluruh komponen Java 2 Platform

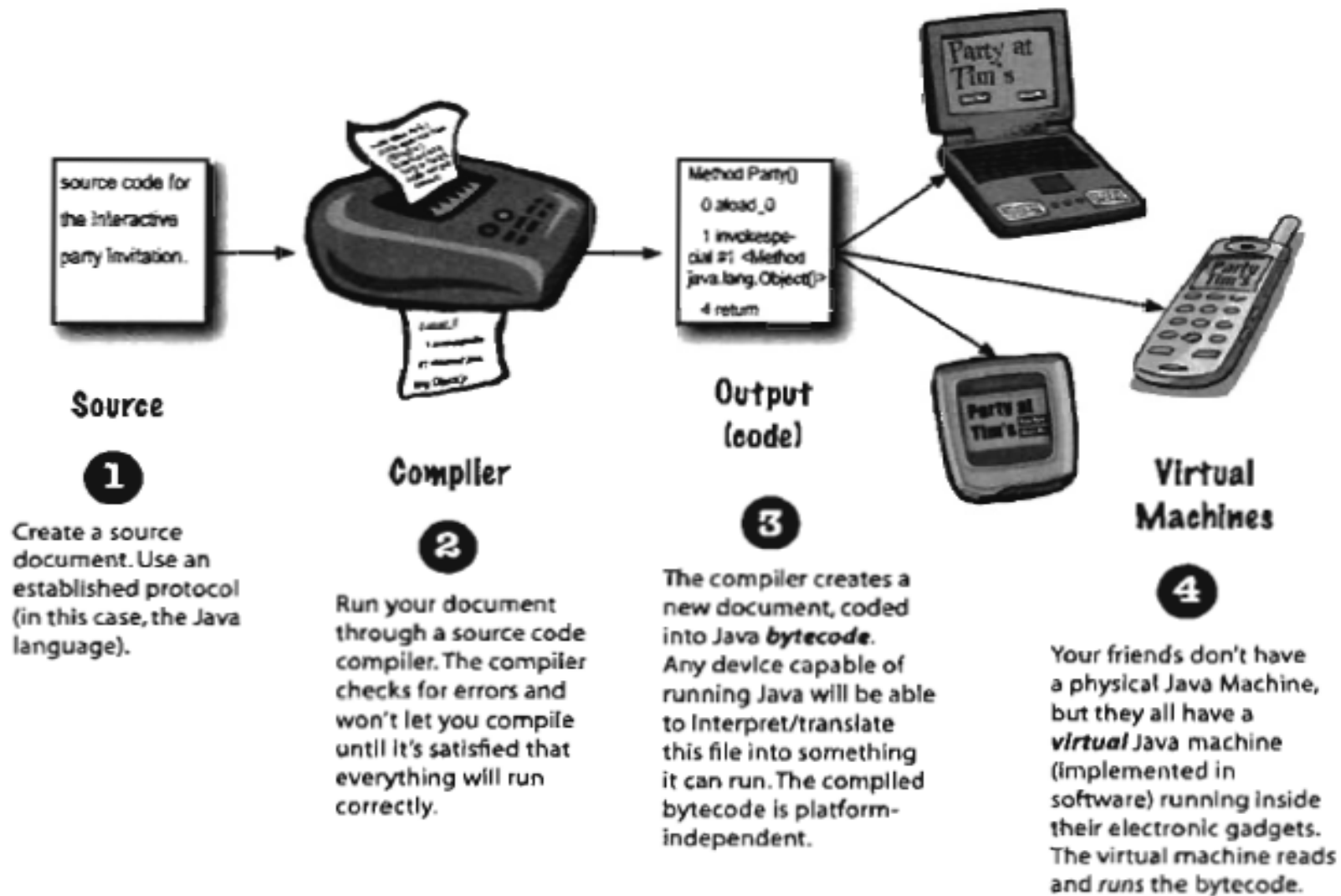
Program Penting pada J2SDK

- Javac -> Compiler
- Java -> Interpreter
- Jdb -> Debugger
- Javap -> Disassembler
- Appletviewer -> Penampil applet
- Javadoc -> Pengenerate documentation
- Javah -> Pengenerate header bahasa C

Proses Kompilasi

- ❑ Program ditulis dengan bahasa Java
- ❑ Pada kompilasi Java, terdapat 2 bagian penting yaitu: Java Compiler dan Java Interpreter
- ❑ Java compiler mengenerate bytecode (sekumpulan instruksi yang me-ressemble kode mesin, tetapi tidak spesifik pada prosesor) yang bersifat “**cross-platform intermediary, halfway between source code and machine language**”.
- ❑ Java interpreter akan meng-interpret Java bytecode yang ada untuk eksekusi program.
- ❑ Kekurangan dari bytecode terletak pada kecepatan eksekusi, karena tidak bersifat **native**

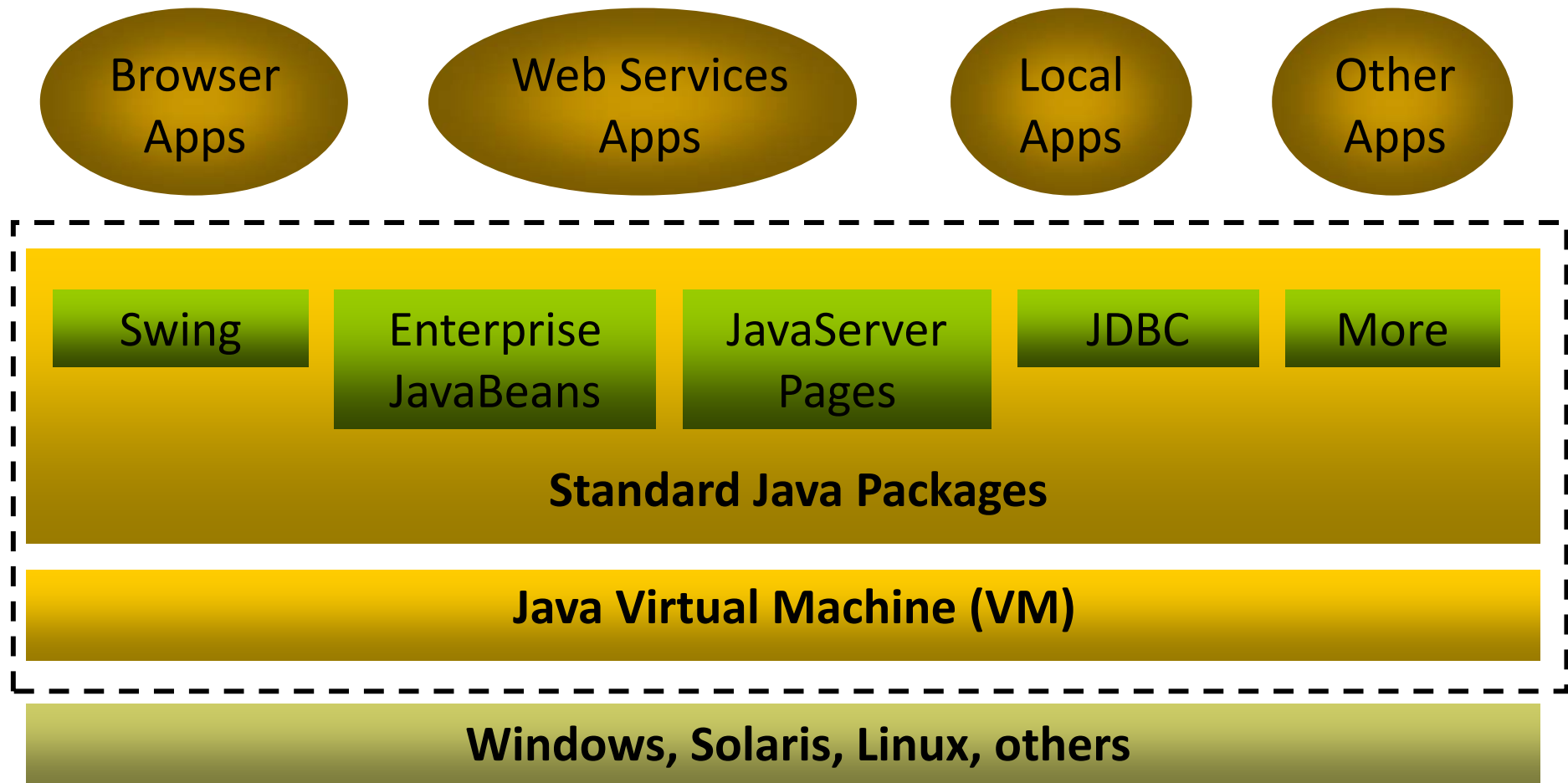
How java works



4 Aplikasi Java

- ❑ **Applications:** program standalone di komputer, dari aplikasi console sampai dengan GUI yang kompleks yang menggunakan **javax.swing**
- ❑ **Applet:** program Java yang dijalankan di web browser dengan menggunakan HTML.
- ❑ **Servlet:** program yang melakukan generating isi webpage namun berjalan di java-enabled web server yang kemudian akan dikirimkan hasilnya ke client.
- ❑ **JSP:** aplikasi web yang berjalan di sisi server.

The Java Environment



Guess what it's doing

```
int size = 27;
String name = "Fido";
Dog myDog = new Dog(name, size);
x = size - 5;
if (x < 15) myDog.bark(8);

while (x > 3) {
    myDog.play();
}

int[] numList = {2,4,6,8};
System.out.print("Hello");
System.out.print("Dog: " + name);
String num = "8";
int z = Integer.parseInt(num);

try {
    readTheFile("myFile.txt");
}
catch(FileNotFoundException ex) {
    System.out.print("File not found.");
}
```

Tugas Praktikum

- Buatlah program konversi Suhu dengan Java
- Buatlah program konversi Nilai dengan Java
- Buatlah program untuk menampilkan n bilangan prima pertama!