



Accelerating Information Technology Innovation

<http://aiti.mit.edu>

Cali, Colombia
Verano 2012

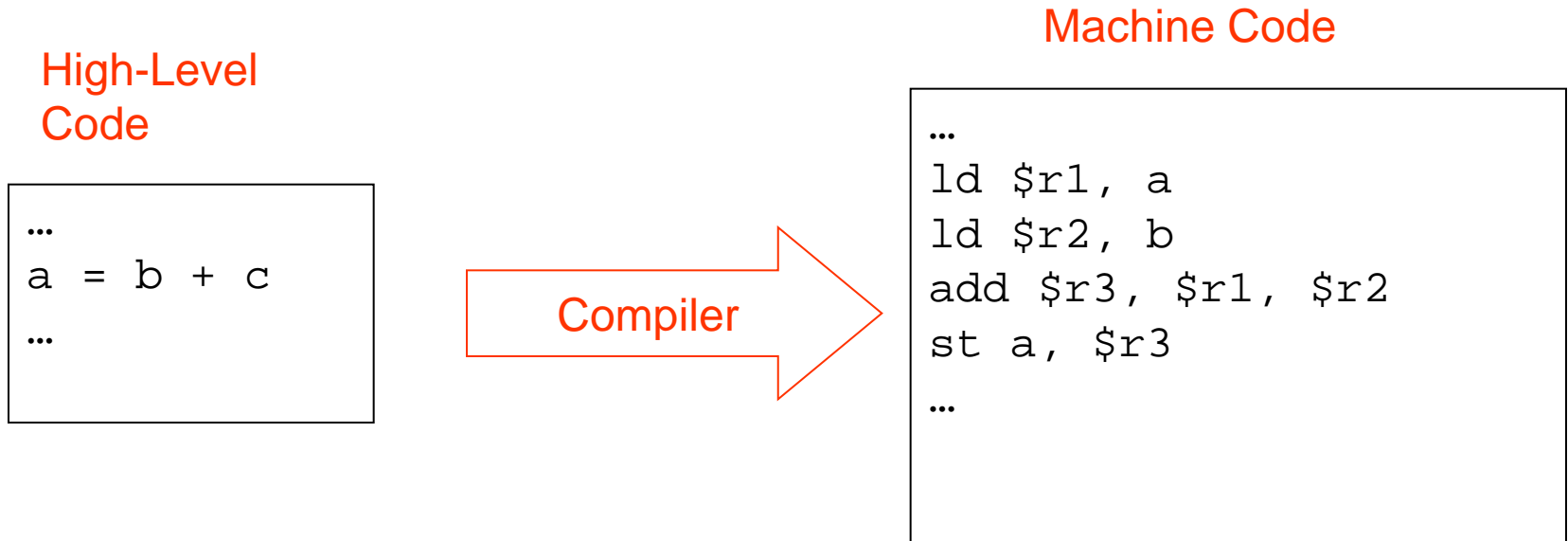
Lección 1 – Introducción a Java

Agenda

- Qué es lo que hace Java especial?
- Ventajas y Desventajas de usar Java.
- Metodología para desarrollar aplicaciones.

Compilador (Compiler)

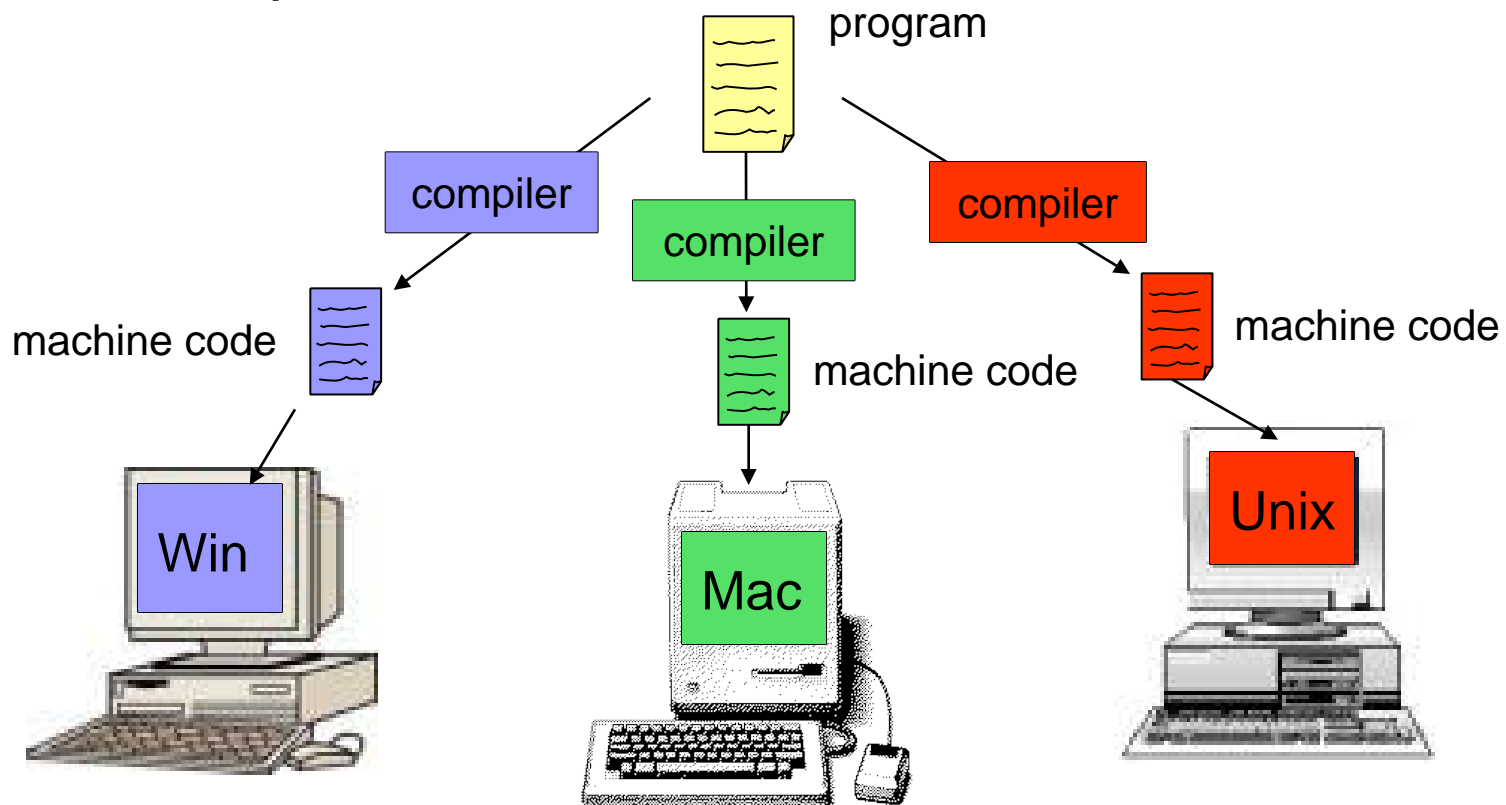
- **Compilador:** Un programa que traduce un lenguaje de programación a código de maquina



- Típicamente se necesita un *compilador* para cada combinación de operador de sistema y computador o maquina - (*plataforma*)

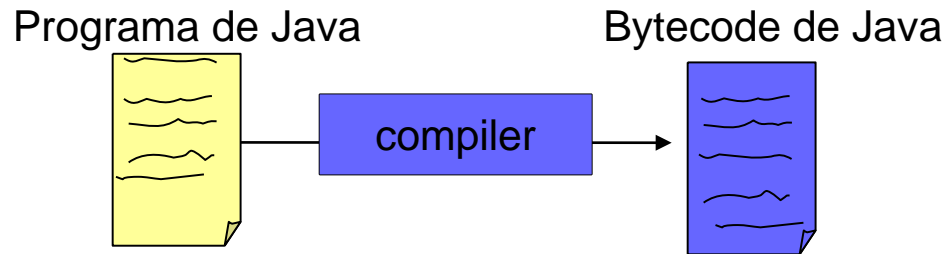
Compilando Programas de Computador

- Diferentes plataformas requieren diferente código de máquina. Por esto, se debe de compilar el programa para cada plataforma, **después** se ejecuta el código de máquina.



El Compilador de Java es Diferente!

- El compilador de Java produce un formato intermedio llamado *bytecode*.

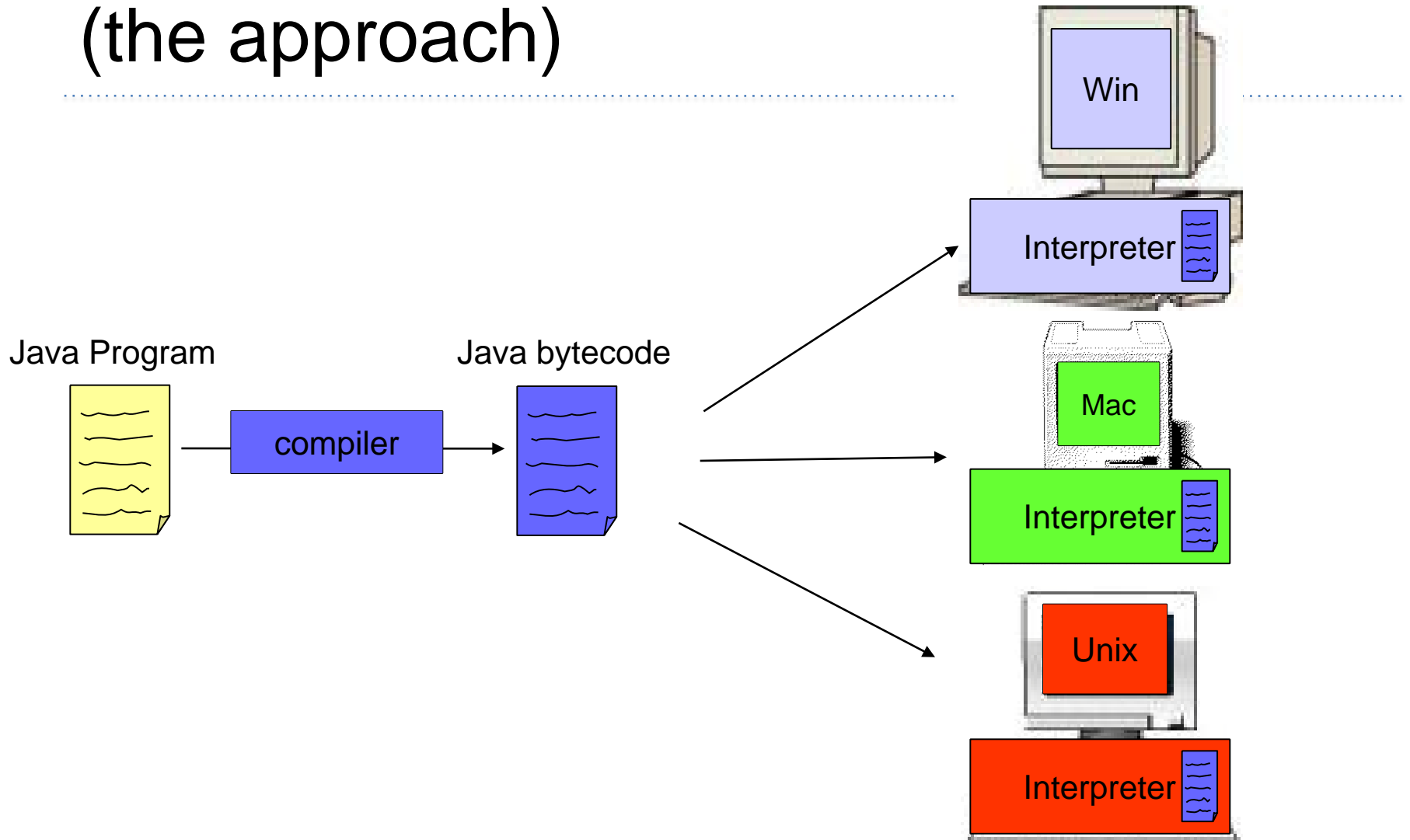


- Bytecode **no es** código de máquina para alguna máquina o computador real.
- Bytecode **es** código de máquina para un computador modelo.
 - Se llama *Java Virtual Machine (JVM)*.

Interpretador de Java

- Un *Interpretador* de Java es requerido para ejecutar el *bytecode* en un computador físico.
- Un interpretador de Java convierte el bytecode a código de máquina (machine code).
 - Al momento que esta ejecutando el programa.
 - *Simula* la ejecución del *Java Virtual Machine* en la computadora física.
- Se puede ejecutar bytecode en cualquier computadora que tenga un interprete de Java instalado (*JRE – Java Runtime Environment*)!
 - Solo se corre por el compilador una vez
 - Se distribuye el mismo bytecode a todos

La Propuesta de Java (the approach)



Ventajas de Java

- Una vez que el programa este compilado se puede ejecutar el bytecode en cualquier dispositivo con un interpretador de Java.
 - Java es *independiente del dispositivo (device-independent)*.
- Java es seguro. El lenguaje de Java y compilador restringe ciertas operaciones para prevenir errores.
 - Quisieras que tu aplicación tomara control total de tu telefono?
 - Hacer llamadas, mandar mensajes (SMS) de texto?
- Java estandariza muchas estructuras y operaciones como listas, manejo de conexiones de interconexiones (network connections) y provee interfaces gráficas de usuario (graphical user interfaces)

Desventajas de Java

- Ejecutar bytecode por un interprete no es igual de rápido que la ejecución directa usando código de máquina
- Usar capacidades específicas del dispositivo (ej. Bluetooth) es en veces difícil porque Java es independiente al dispositivo (device-independent)
- Para poder ejecutar un programa de Java en múltiples dispositivos, cada uno requiere un interprete de Java
 - Ej: Mayoría de celulares Nokia contiene un interprete de Java

Metodología de Programación

1. Especifica y analiza el problema
 - Remover ambigüedad
 - Decide en entradas/salidas y algoritmos
2. Diseña la solución del programa
 - Organiza el problema en piezas pequeñas
 - Identifica código existente para reusar!
 - Documento de diseño
3. Implementación (programación)
4. Probar y verificar la implementación (test and verify)
5. Mantener al corriente el programa (mantenimiento)



Escribir buen código

- Un programa que cumple con las especificaciones no es necesariamente un buen programa.
- Podrás hacer cambios al programa?
 - Entenderás TU el código después de algún tiempo?
- Otros tendrán que ver tu código
 - Lo entenderán ellos?
- Escribe tu programa para que sea facil de entender y sea extensible!
 - Toma tiempo extra para pensar en estas cosas.

Ejemplo de Código: Comentarios

```
/* The HelloWorld class prints "Hello,
World!" to the screen */
public class HelloWorld {
    public static void main(String[] args) {
        // Prints "Hello, World!"
        System.out.println("Hello, World!");
        // Exit the program
        System.exit(0);
    }
}
```

Comentarios (comments)

- *Comentarios* son usados para describir el comportamiento del código
 - Asiste a otros y a ti mismo a leer el código
 - El compilador Java ignora los comentarios.
- Comentarios son hechos con `//` y con `/* */`
- Dos ejemplos:
 - `/* The HelloWorld class prints "Hello, World!" to the screen */`
 - `// Prints "Hello, World!"`

Comentarios sobre Comentarios

- Puedes colaborar en proyectos de software con gente de otras partes del mundo y nunca conocerlos personalmente
- Deben de poder entender como trabaja tu código solo con leer los comentarios
- Cualquier cosa que no sea intuitiva o claramente evidente requiere un comentario
- 50% de tu código puede ser comentarios
- Codificar es fácil; escribir buenos comentarios no lo es