Programmazione II

A.A. 2022-23

Prof. Maria Tortorella

Traduttori
Programmazione object-oriented e Linguaggio di programmazione Java

Dal linguaggio di alto livello al linguaggio di basso livello

IDEA: scrivo le mie istruzioni usando un linguaggio a me più comprensibile e poi le *traduco* in linguaggio macchina.

Traduttori

Servono a generare software.

Generano codice in *linguaggio macchina* a partire da codice scritto in un linguaggio di programmazione ad alto livello (ad es. C++, Java).

Si distinguono in:

interpreti,

compilatori.

Compilatori

Un compilatore è un programma che prende in input un codice *sorgente* e lo traduce fornendo in output un codice *oggetto*.

Per eseguire un programma *sorgente P*, scritto in un linguaggio di programmazione *L*:

P viene tradotto in un programma *Q* equivalente scritto in linguaggio macchina;

il programma *Q* viene eseguito.

Esempi di linguaggi compilati:

C++, Pascal, Cobol, Fortran, …

Il compilatore è legato all'architettura della macchina.

Interpreti

Un interprete è un programma che prende in input un codice sorgente e, passo dopo passo, traduce ed esegue ogni singola istruzione.

- La traduzione avviene dunque simultaneamente all'esecuzione.
- Per ogni istruzione del programma sorgente *P*:
  - Viene tradotta la *singola* istruzione generando il corrispondente insieme di istruzioni in linguaggio macchina;
  - Si esegue il codice in linguaggio macchina e si passa all'istruzione sorgente successiva.

Esempi di linguaggi interpretati:

- VBasic, Lisp, Prolog, Java (inizialmente).

Compilatori vs. Interpreti

Interpreti
- Lenti nell'esecuzione.
- Spesso si utilizza un linguaggio intermedio.
- Progetti di dimensione limitata.
- Facilità d'interazione col codice e velocità di sviluppo.
- Facili da scrivere.

Compilatori
- Veloci nell'esecuzione.
- Necessitano di poca memoria.
- Permettono la compilazione separata.
- Difficili da scrivere.

Il linguaggio Java

Il linguaggio di programmazione Java è:

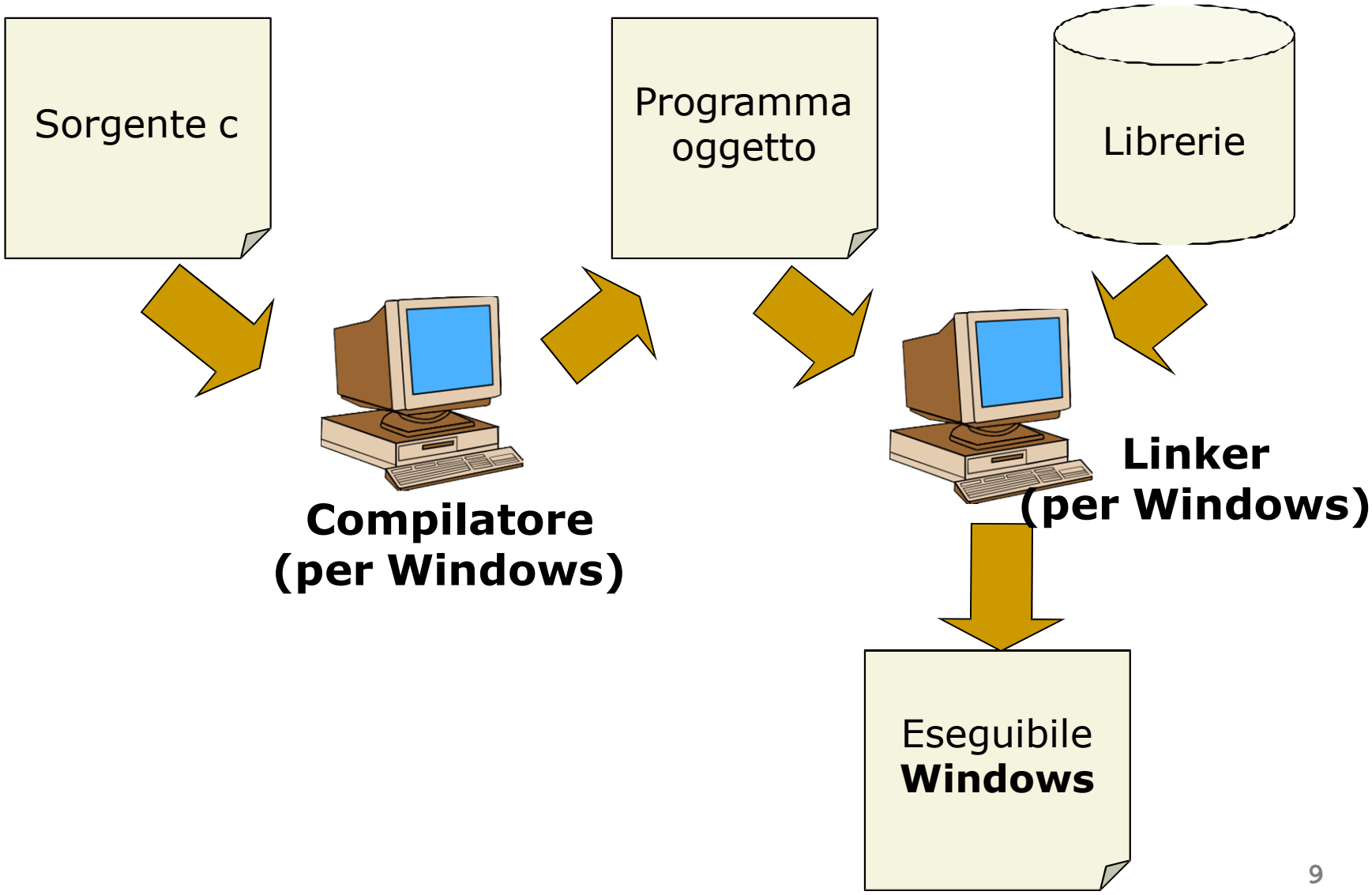
- Semplice
- Sicuro
- Indipendente dalla piattaforma
  - "write once, run anywhere"
- Libreria molto ricca
  - packages
- Pensato per Internet

Riprendiamo il processo di traduzione

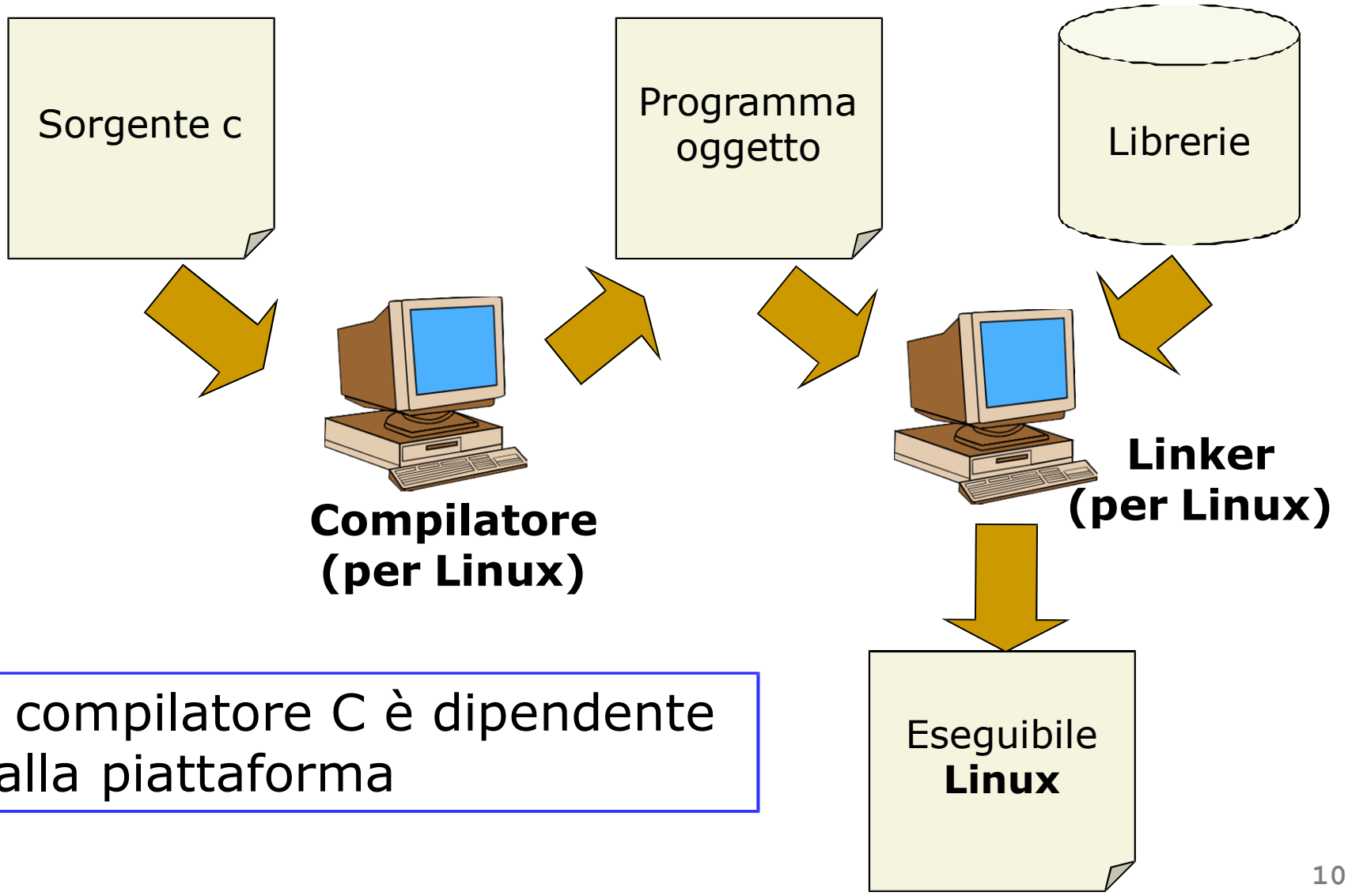
Traduttori

- Servono a generare software che può essere eseguito.
  - Generano codice in *linguaggio macchina* a partire da codice scritto in un linguaggio di programmazione ad alto livello (ad es. C++, C, Pascal).
- Si distinguono in:
  - interpreti,
  - compilatori
- Ne abbiamo già analizzato velocemente le differenze

Traduzione C



Traduzione C



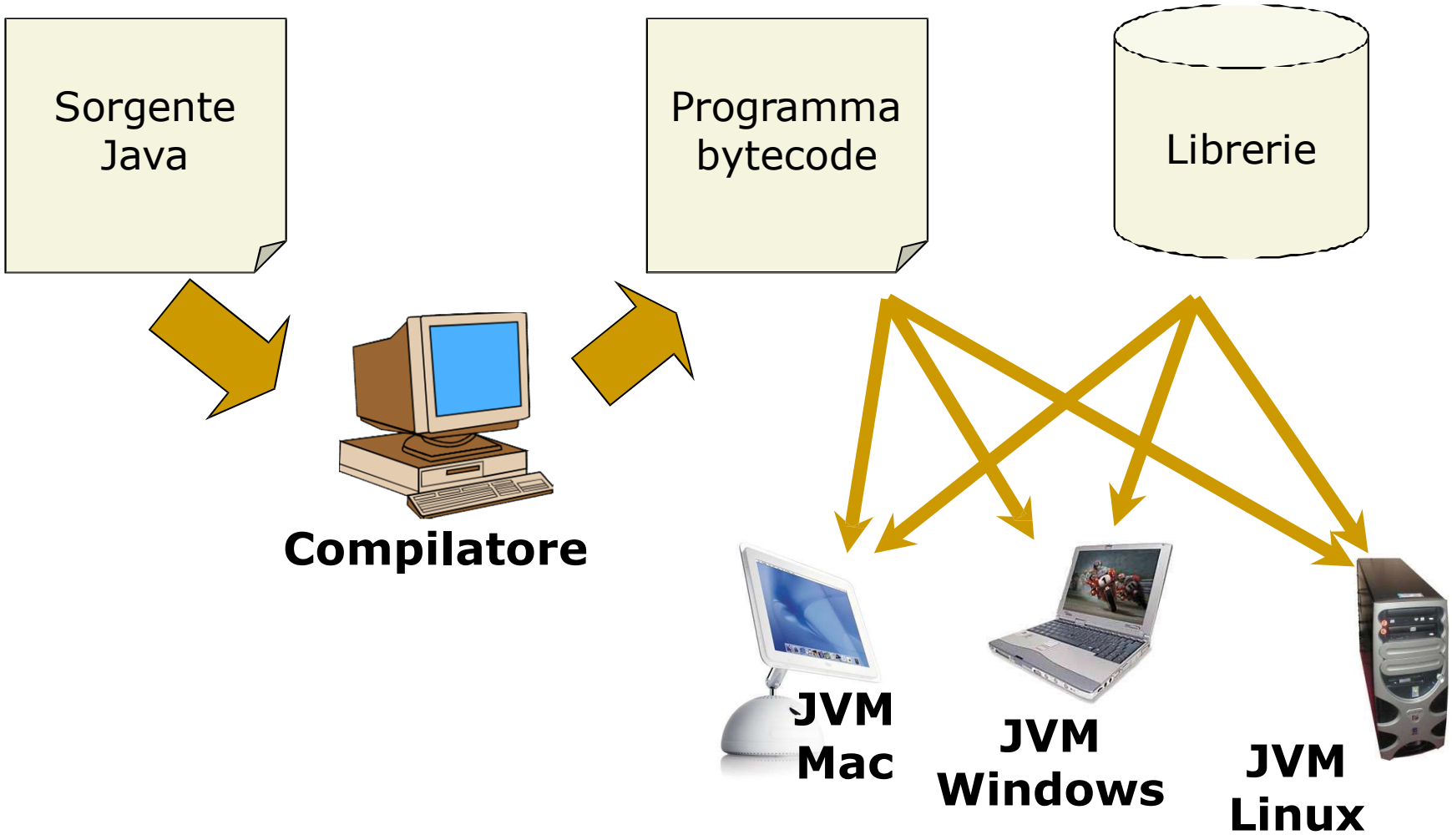
Il compilatore C è dipendente dalla piattaforma

Traduzione Java

- Sfrutta sia la compilazione che l'interpretazione
- Esiste un codice intermedio
  - **Java Byte Code**
    - **Java Virtual Machine**
    - Il linguaggio macchina di una macchina virtuale
  - NON leggibile dall'uomo
  - Conservato in **class files**

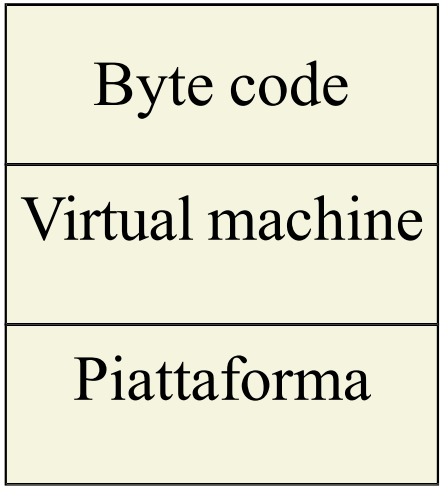
- Il compilatore Java non produce un programma scritto in linguaggio macchina ma in Java Byte Code
- Il programma prodotto deve essere interpretato (Java Virtual Machine) per essere eseguito

Traduzione Java



Java Virtual Machine

- Un programma Java NON può essere eseguito senza Java Virtual Machine
  - Interpreta il codice Java Byte Code
  - È a sua volta in esecuzione sul calcolatore



Programmazione object-oriented

Concetti principali:
Classe, oggetto e messaggi

Linguaggi orientati agli oggetti

- Forniscono astrazioni che consentono di rappresentare direttamente nel dominio della soluzione gli elementi del dominio del problema
  - Oggetti
  - Classi
  - Messaggi

Astrazione

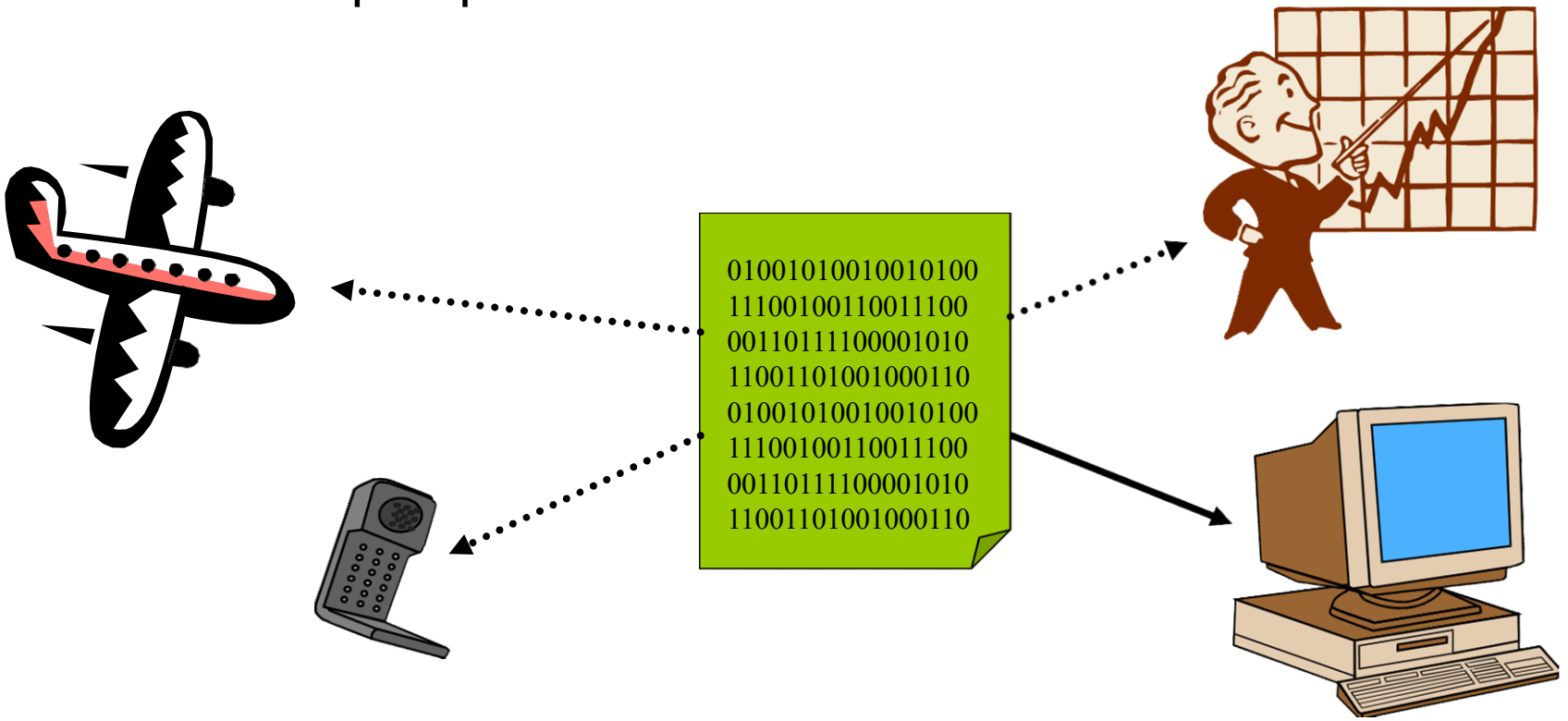
Abstraction: A view of an object that focuses on the information relevant to a particular purpose and ignores the remainder of the information

Information Hiding: A software development technique in which each module's interfaces reveal as little as possible about the module's inner workings and other modules are prevented from using information about the module that is not in the module's interface specification

- IEEE standard glossary of software engineering terminology

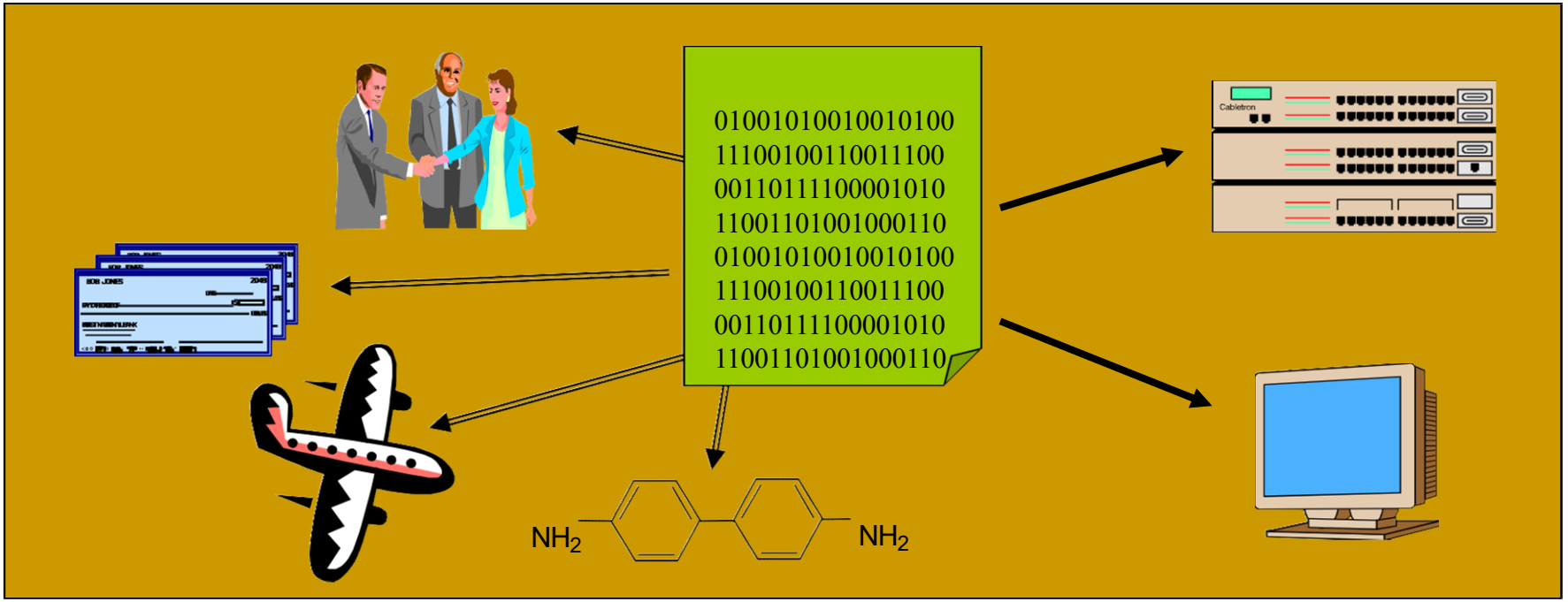
Programmi e mondo reale

- Il codice di un programma interagisce con il calcolatore ed i suoi componenti per realizzare delle soluzioni per processi esterni



I programmi come modelli

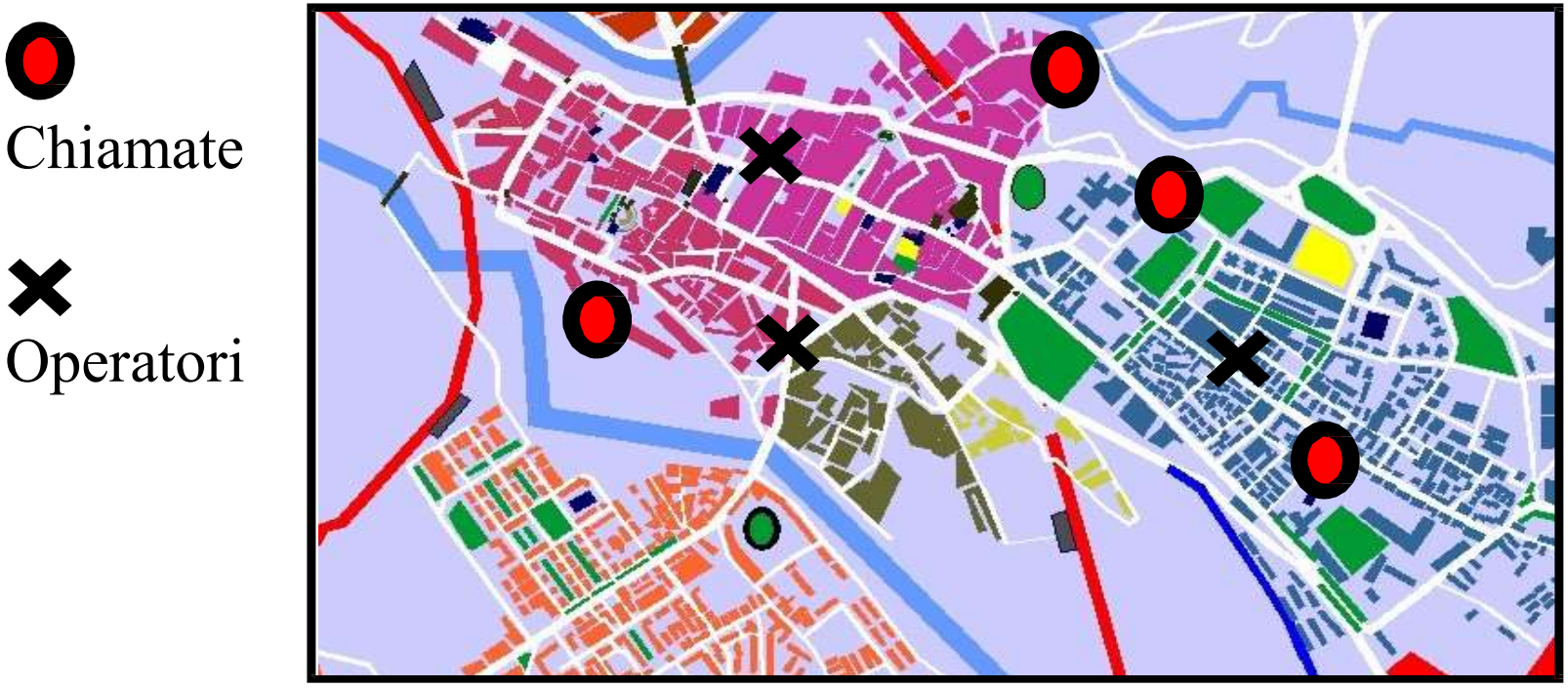
- Il codice di un programma modella entità reali
  - Interne: tastiera, dischi, reti, schermo
  - Esterne: piani di volo, molecole, transazioni bancarie



I modelli di permettono di creare una rappresentazione del problema da risolvere che ignora i dettagli implementativi e si concentra solo sugli aspetti principali del problema dal risolvere

Modelli

- I modelli sono nati prima dei calcolatori e NON devono necessariamente essere realizzati mediante calcolatori



Elementi del modello

- Ogni modello è formato da elementi che rappresentano entità
- Gli elementi del modello presentano un comportamento consistente con il modello
- A seconda dei loro comportamenti comuni, gli elementi possono essere raggruppati in categorie diverse
  - Gli elementi della stesso categoria sono elementi dello stesso tipo e svolgono lo stesso compito
- Il comportamento di un elemento può essere provocato da azioni esterne

Modelli nella progammazione Object Oriented ed in Java

- Elementi del modello: Oggetti
- La categorie di oggetti vengono chiamate Classi
- Una classe:
  - Determina il comportamento degli oggetti appartenenti
  - E' definita da una sezione di codice
- Un oggetto
  - Appartiene ad una classe
  - Costituisce una **istanza** di tale classe

Esempio

- Classe operatore:
  - Definisce il comportamento degli operatori (ad esempio, cambiamento di locazione, registrano il tempo di un intervento, ecc.)
  - Ogni operatore in servizio è una istanza di tale classe
- Classe chiamata:
  - Definisce il comportamento delle chiamate (ad esempio, stabilisce una priorità, viene fatta in un dato orario, viene eseguita da cliente chiamante ecc.)
  - Per ogni chiamata che arriva si crea una istanza

Programmazione OO

- Fuoco: gli oggetti
  - e le classi che ne definiscono il comportamento
- Filosofia: In un programma in esecuzione sono gli oggetti che eseguono le operazioni desiderate
- Programmare in Java
  - Scrivere le definizioni delle classi che modellano il problema
  - Usare tali classi per creare oggetti
- Java è dotato di classi ed oggetti predefiniti
  - Non si deve continuamente reinventare le ruota

Classi ed oggetti