La cassetta di un buon carpentiere del software

Luca Amato

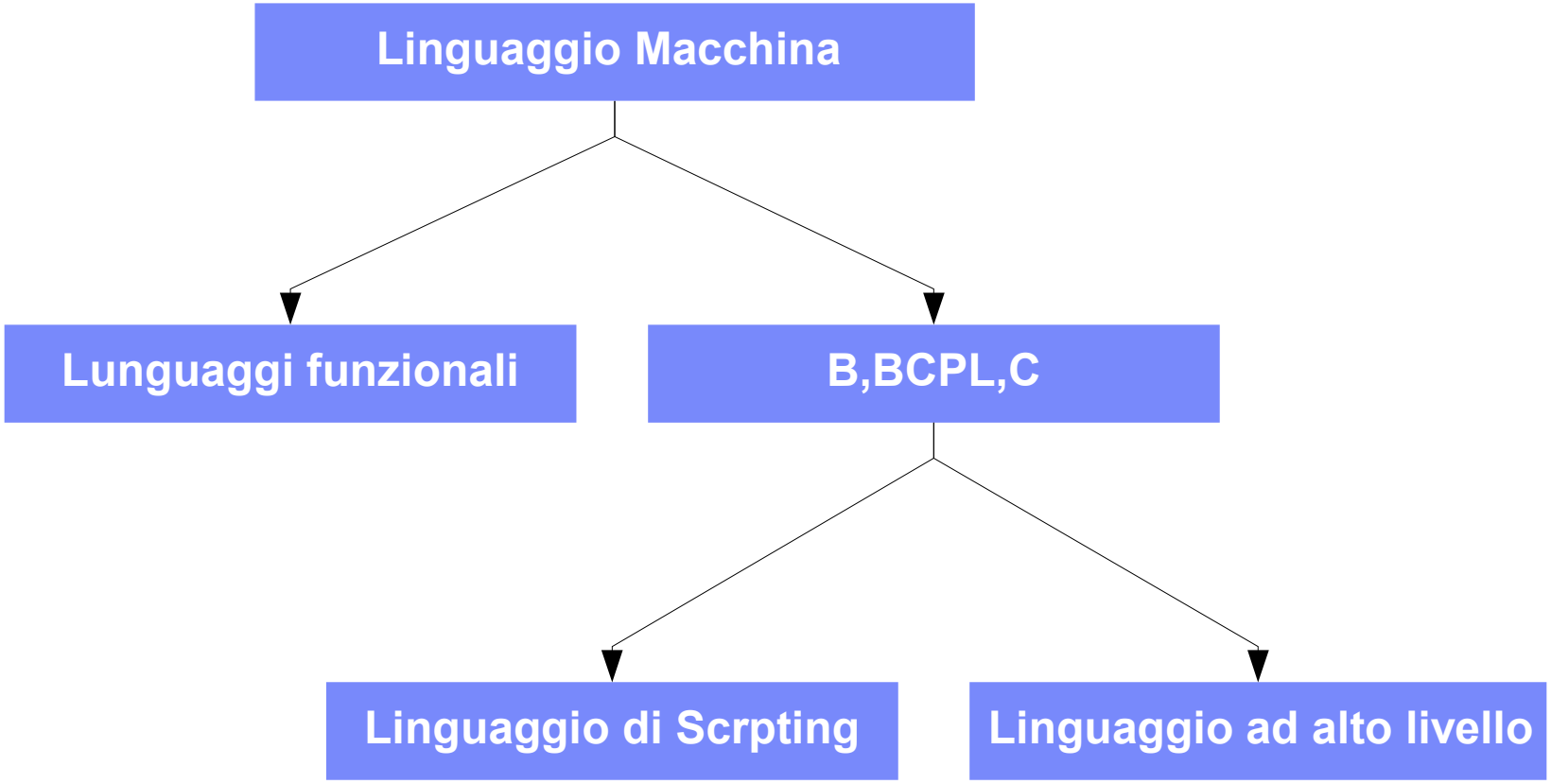
lucaamato@it.ibm.com

ON DEMAND BUSINESS™

Agenda

- Microstoria dell'informatica e dei linguaggi
- Scegliere un linguaggio di programmazione
- Il grande sogno: “write once/run anywhere”
- La vera programmazione multipiattaforma

Micro storia dell'informatica



Linguaggi funzionali

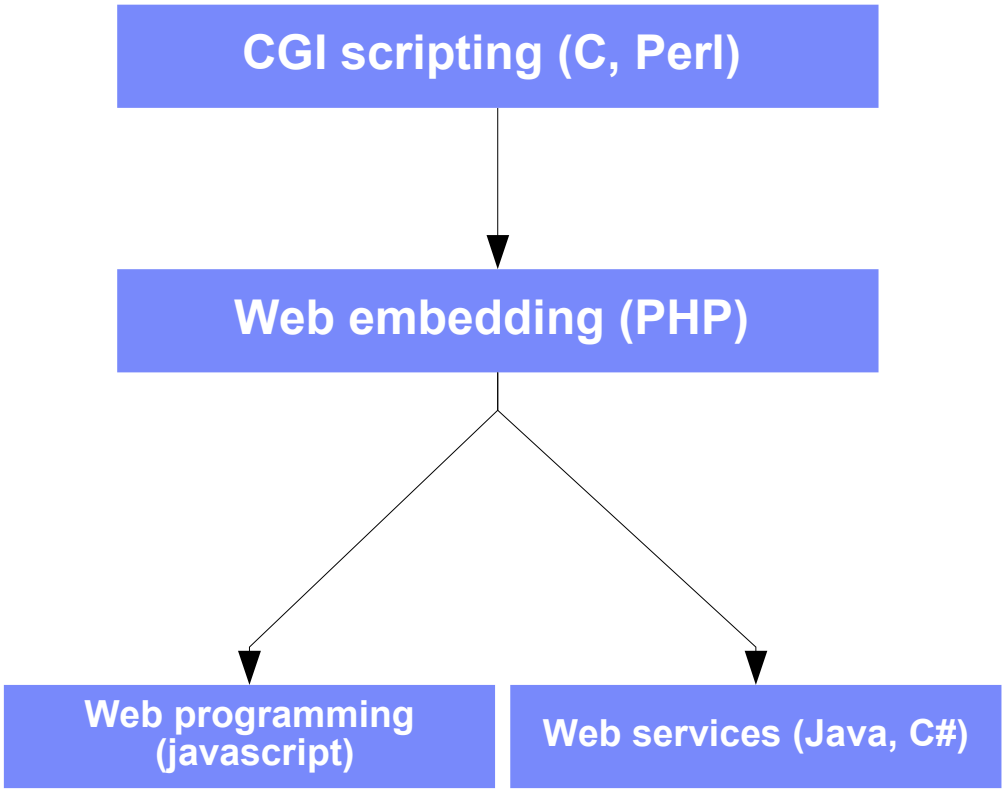
- **Linguaggi della prima generazione**
  - **In genere erano compilati**
  - **LISP, Scheme**
- **Linguaggi funzionali moderni**
  - **Interpretati oppure compilati**
  - **Includono estensioni per la programmazione OO**
  - **In genere vengono compilati**
  - **LISP, Scheme, Haskell, ML, ...**
  - **Streamlined composite application management**
  - **Direct-to-operations productivity**

Linguaggi OO

- **Smalltalk, Objective-C, C++, ..., Java, C#**
  - **Modularizzazione del codice**
  - **Data hiding (migliore definizioe delle API)**
  - **Ereditarietà, classi, istanze, interfacce**

La programmazione ormai OO è avunque; dal LISP al Pascal, dal Perl al Python, dal PHP a JAVA

Mondo Web e anche non-Web



- Sistemi operativi
  - Librerie di base C
- Interfacce grafice (C, C++, Java)
  - Applicativi server
  - Applicativi client
  - Wrapper
- Scripting (Bash, Perl, Python)

La cassetta degli attrezzi

Cosa serve come minimo, ad un buon "carpentiere" del software

- **Un linguaggio di basso livello**
- **Un linguaggio di scripting**
- **Un linguaggio di alto livello**
- **Un linguaggio per il web (server-side)**
- **Un linguaggio per il web (client-side)**
- **...e poi ancora qualcosa?**

Alcune regole semplici

- Abbandonare i linguaggi di embedding web
  - (per prima cosa uccideremo il PHP)
- Il C è l'unico linguaggio binding-friendly
- Se possibile usare linguaggi inter-operabili
  - Usare il C++ significa C++ per sempre
- Se possibile scegliere linguaggi definiti come standard, con più di un'implementazione
  - Javascript è meglio di ActionScript
- E' ora di cominciare a fidarsi delle VM

Linguaggi di scripting

Python, ruby, Perl 5 & co.

- Interpretati (JIT non disponibili o scadenti)
- Completa libreria di base
- Tipizzazione dinamica
- Estensioni funzionali
- Wrist-friendly
- Sviluppo estremamente veloce
- Molte eccezioni a run-time
- Strumenti di debug ancora immaturi

I vantaggi delle VM

Utilizzare linguaggi che vengono compilati per Virtualmachine hanno molti vantaggi

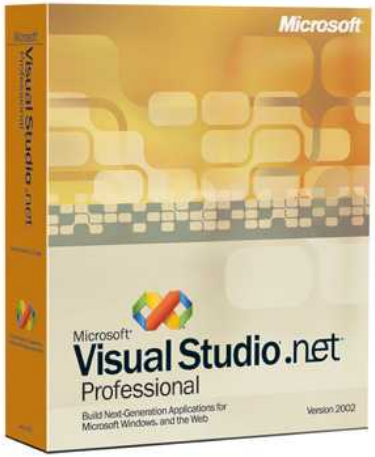
- Con l'hardware disponibile qusi nessun protocollo richiede più l'ottimizzazione manuale del codice
- I JIT fanno un ottimo lavoro (1x-5x del C)
- Il byte code è davvero multipiattaforma
- Il codice è, prese le ***ovvie e dovute precauzioni*** multi-piattaforma
- Il codice è in genere più sicuro
  - Difficile dare un definizione di “sicuro”

Il linguaggio Java

- Java (nell'implementazione di SUN)
  - Linguaggio propietario
  - Poche estensioni funzionali
  - Tipizzato staticamente
  - JIT estremamente veloci e efficenti
  - Ampia libreria di base
- Kaffe ed altre implementazioni libere
  - Linguaggio propietario, implementazione llibera
  - Libreria di base ancora manchevole

E Microsoft ?

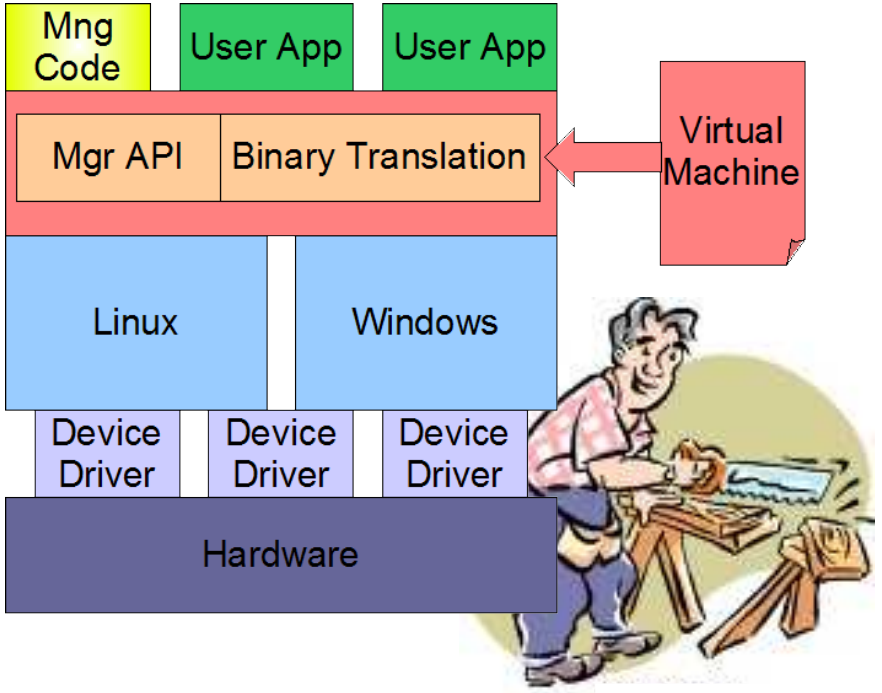
- **La Common Language Infrastructure**
  - **Definita come standard dall'Ecma**
  - **C#, VB.NET, JS e C++**



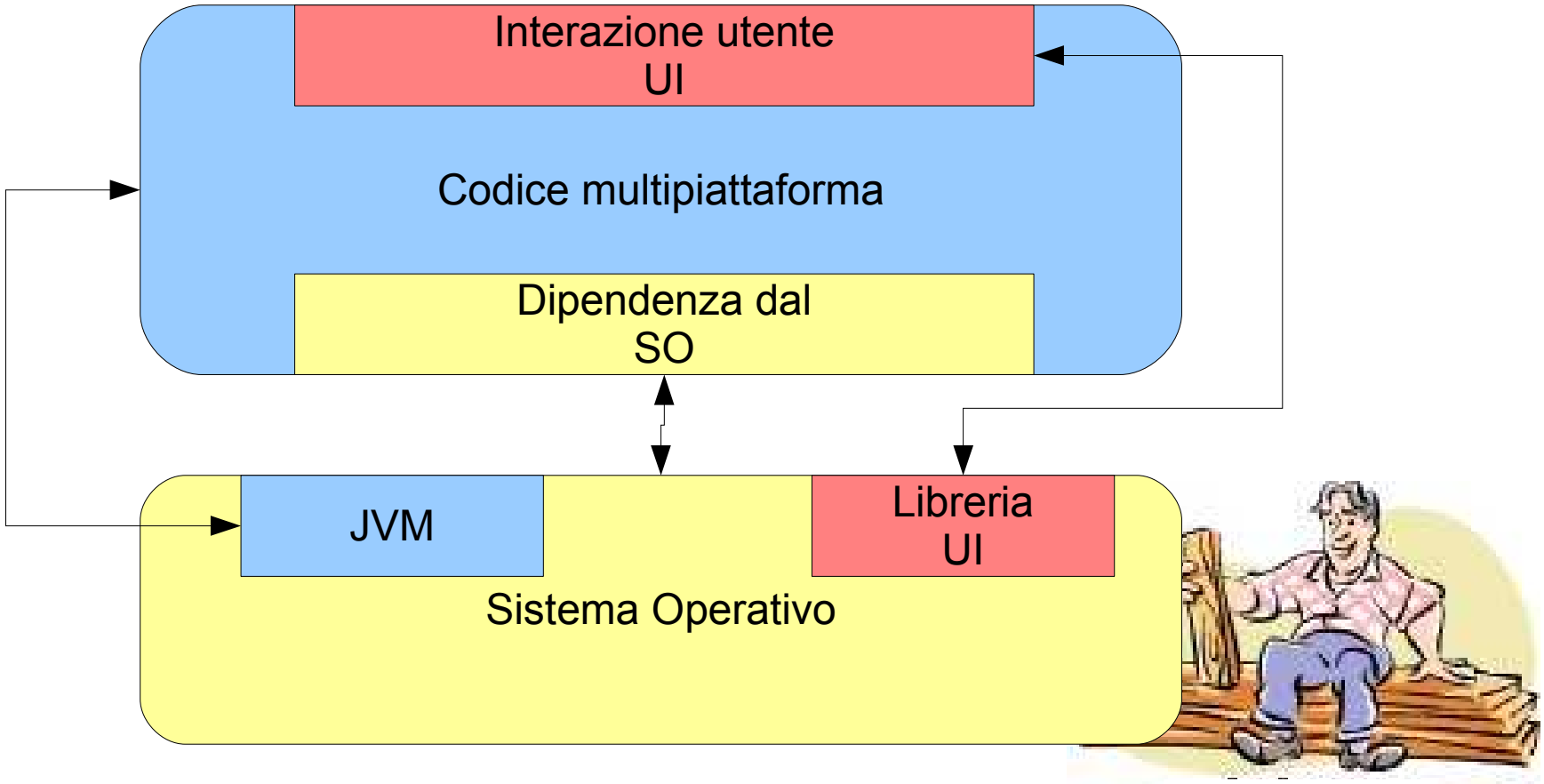
Il mito: write-once, run-anywhere

Ogni programmatore sogna di scrivere codice che giri su qualsiasi piattaforma

- **Java permette da anni di realizzare almeno in parte queto sogno**
- **Le JVM di fatto permettono già al 90% del mio codice di essere platform independent**
- **L'overhead pagato dalle JVM è sopportabile**
- **JIT realizzano codice della stessa efficienza del codice compilato (e qualche volta superiore)**



Un sogno piu realistico



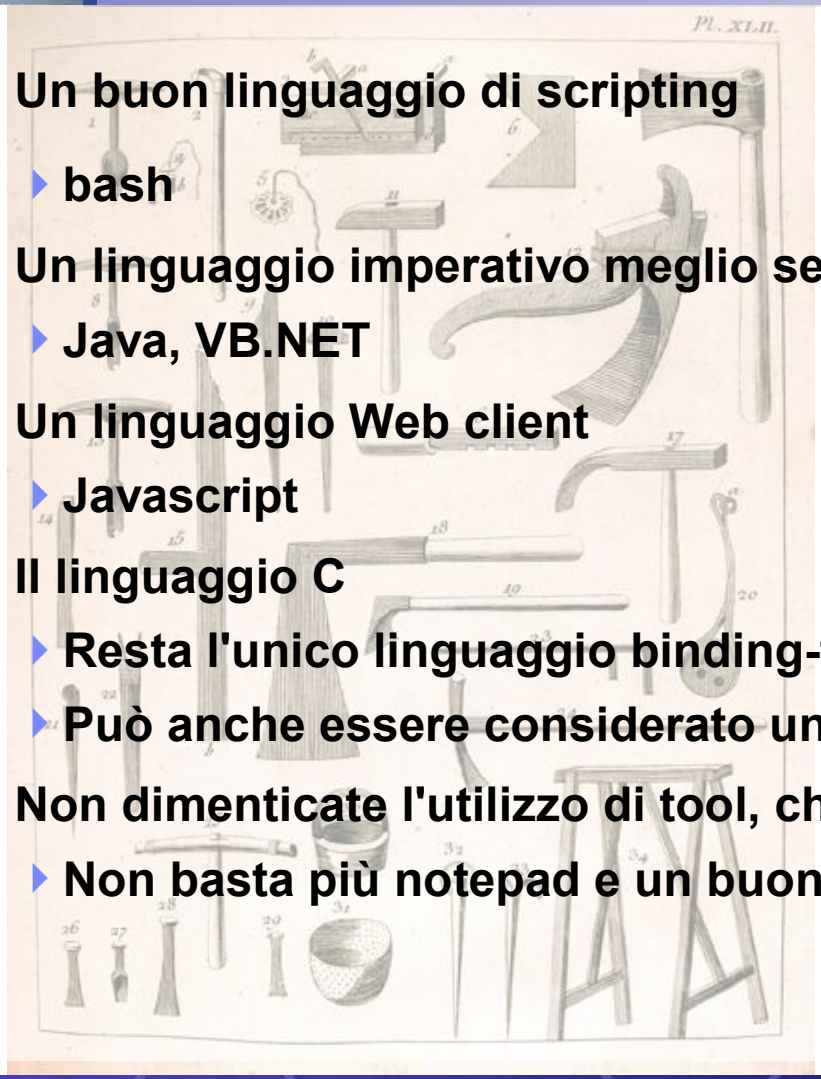
Una VM per legarli

Una sola VM, in grado di chiamare direttamente il C senga glue-code

- Chiamate dirette al sistema operativo ed al codice legacy
- Prototipizzazione rapida con linguaggi agili
- Utilizzo di un linguaggio type-safe
- Utilizzare un mix di linguaggi per affrontare diverse problematiche con lo strumento piu adatto

Conclusioni

- Un buon linguaggio di scripting
  - bash
- Un linguaggio imperativo meglio se type-safe
  - Java, VB.NET
- Un linguaggio Web client
  - Javascript
- Il linguaggio C
  - Resta l'unico linguaggio binding-friendly
  - Può anche essere considerato un linguaggio a basso livello
- Non dimenticate l'utilizzo di tool, che sono diventati ormai indispensabili
  - Non basta più notepad e un buon compilatore per essere produttivi



Thank You

Luca Amato

lucaamato@it.ibm.com

ON DEMAND BUSINESS™