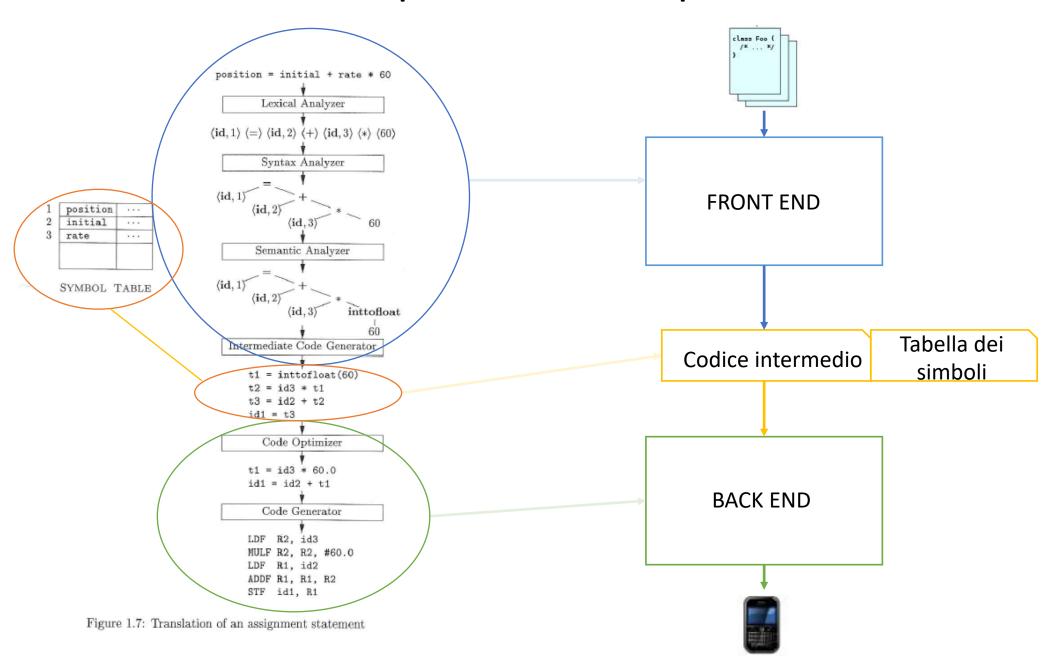
Infrastrutture per la compilazione ed esecuzione dei linguaggi di programmazione

Compilatore a due passi

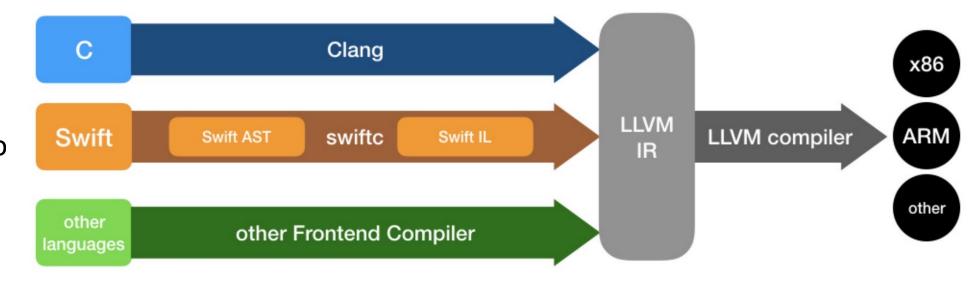


Compilatori multipiattaforma e multilinguaggi

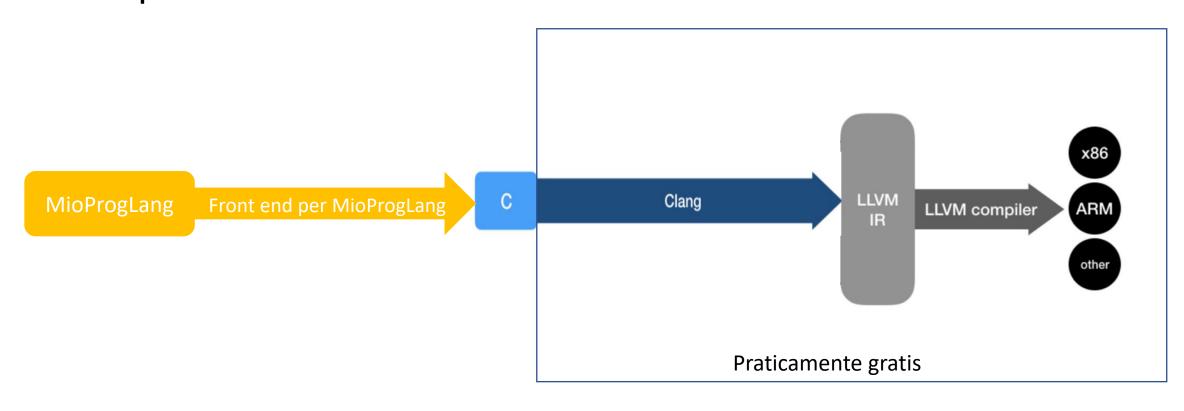
- Fissa un codice intermedio e rendilo noto a chi scrive front end ed a chi scrive back end
- Chi scrive il front end?
 - Chi ha un linguaggio di programmazione (sorgente) da implementare
 - Non deve preoccuparsi delle specifiche del linguaggio macchina target
- Chi scrive il back end?
 - Chi ha un'architettura hardware su cui vuole far eseguire programmi
 - Non deve preoccuparsi delle specifiche del linguaggio di programmazione sorgente

Codice intermedio LLVM IR

- Parte della LLVM Compiler Infrastructure (http://llvm.org)
- Linguaggi che compilano in LLVM IR:
 - C/CLang
 - C++
 - Swift
 - Rust
 - Kotlin nativo



Molteplici codici intermedi per un compilatore

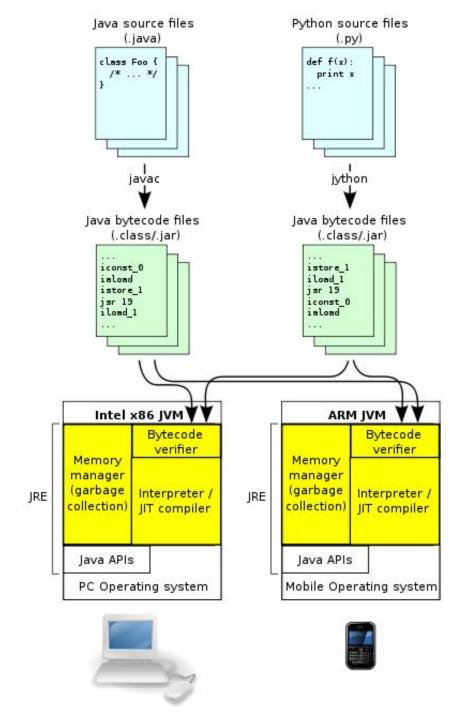


Infrastrutture ibride

- Quelle più note sono Java e .NET, che presentano rispettivamente i seguenti codici intermedi
 - Bytecode
 - Common Intermediate Language (CIL)
- Il back-end è eliminato dal compilatore e sostituito da una macchina virtuale (interprete software) resa più efficiente da un JIT compiler
 - In pratica un JIT compiler implementa parti del back-end che però in questo caso sono eseguite durante l'esecuzione del codice intermedio (a run-time).
 - La macchina virtuale ed il suo jit compiler sono sempre specifici ad un'architettura hardware (come d'altronde il back-end)

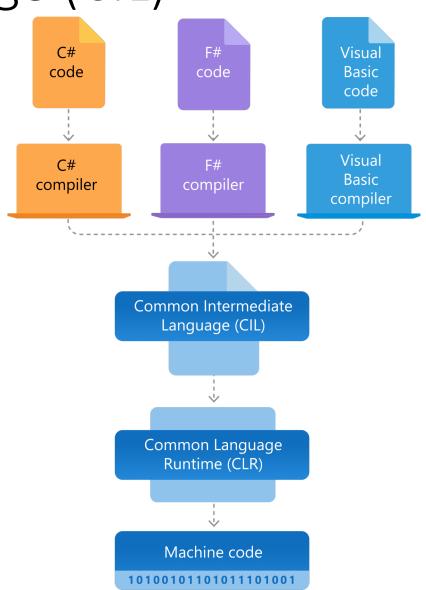
Bytecode

- Eseguito da Java Virtual Machine (JVM) che contiene un JIT compiler
- Linguaggi che compilano in bytecode
 - Java
 - Python
 - Kotlin
 - Clojure,
 - Groovy
 - Scala
 - (vedi https://en.wikipedia.org/wiki/List_of_J VM_languages)



Common Intermediate Language (CIL)

- Parte della infrastruttura .NET
- Eseguito dal CLR che contiene un JIT compiler
- Linguaggi che compilano in CIL
 - C#
 - F#
 - Visual Basic
 - C++
 - (vedi https://en.wikipedia.org/wiki/List_of_CLI_lan guages)



Dart: The platforms

Dart's compiler technology lets you run code in different ways:

• Native platform: For apps targeting mobile and desktop devices, Dart includes both a Dart VM with just-in-time (JIT) compilation and an ahead-of-time (AOT) compiler for producing machine code.

 Web platform: For apps targeting the web, Dart includes both a development time compiler (dartdevc) and a production time compiler (dart2js). Both compilers translate Dart into JavaScript.

II Iinguaggio DART

https://dart.dev/ overview

