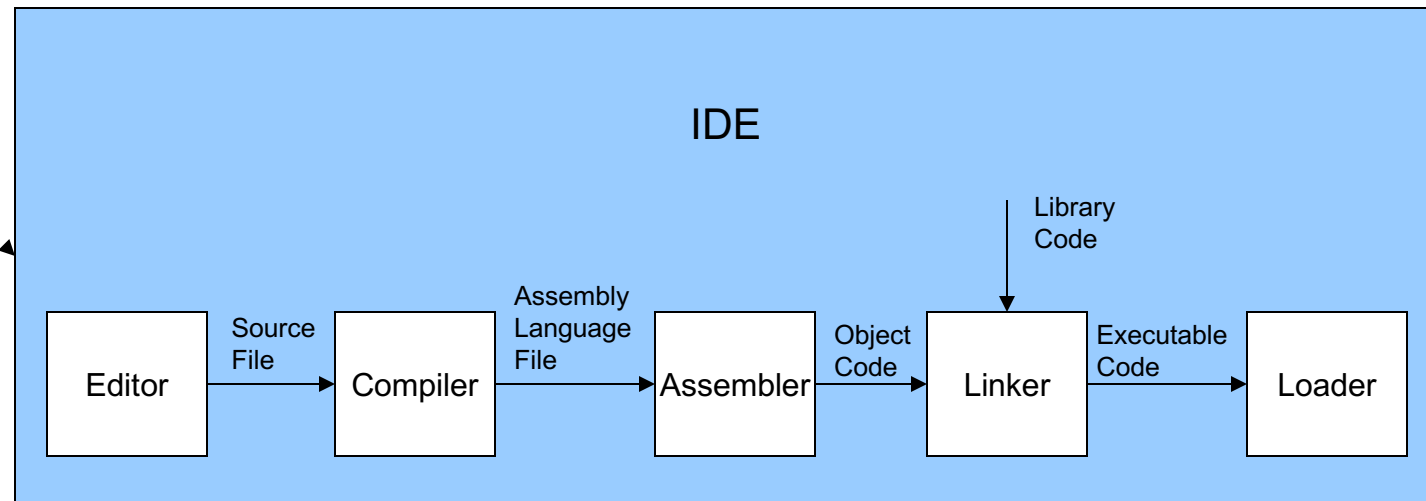


# Language Systems

What actually happens in your IDE? IDE  $\equiv$  Integrated Development Environment

Classical Sequence: C++, C, Fortran

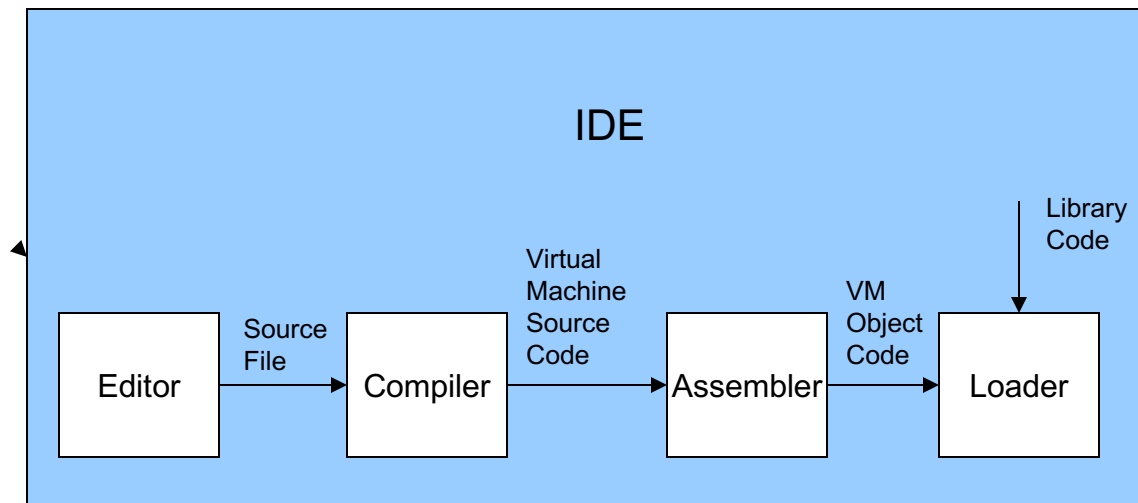
Read Chap 4



# Language Systems

Alternatives, depending on Source Language

Java, C#

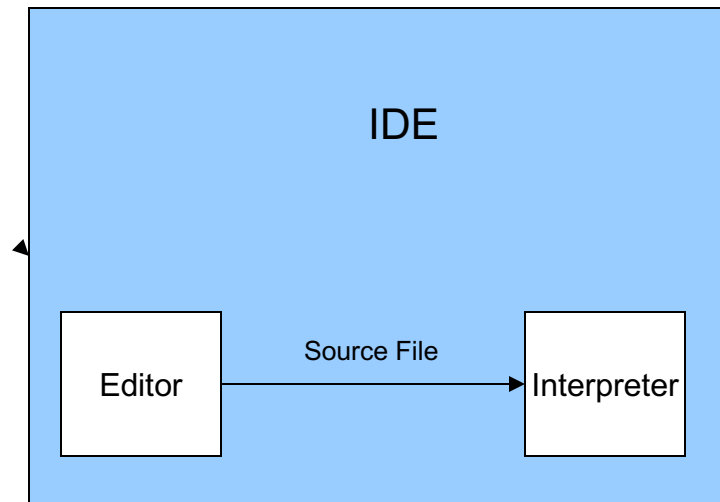


Note: In Java and C# there is no Linker, the program is linked at runtime.

# Language Systems

Alternatives, depending on Source Language

Lisp



# Take Away

- The complexity of the language systems depends on the underlying requirements of the source language itself (i.e. linker, loader, etc. required)
- Typically the IDE's reflect this.

# Observation

Some IDEs have compilers, some have interpreters –  
What is the difference?

- Compilers translate high-level languages (Java, C, C++) into low-level languages (Java Byte Code, assembly language).
- Interpreters execute high-level languages directly (Lisp).

**Note:** Virtual machines can be considered interpreters for low-level languages; they directly execute a low-level language without first translating it.

# Language Systems

- Why choose compilation over interpretation?
  - Compilers can generate very efficient code and, consequently, the compiled programs run faster than interpreted programs.