PRINCIPLES OF PROGRAMMING LANGUAGES
Paradigms of Programming Languages

prof. Marco Bellia, Dip. Informatica, Università di Pisa

February 18, 2013

Overview

- Why and Aims
- Prescriptive vs. Descriptive
- Programming Languages Paradigms
- 50 years of Programming Languages
- Overview of the Module

Why and Aims

- **Why Study Programming Paradigms?**
  To have the right knowledge of the fundamental tools (i.e. Languages) for developing computer applications.

- **Aims**
  - **Choose the most appropriate** language for any given task;
  - **Understand obscure** features of current languages;
  - Have the right competence **for learning the novel** Programming Languages (syntax, semantics, implementation, and use) of the next future

Prescriptive vs. Descriptive

- Remember:
  - Algorithms = Logic + Control
  - Programs = A Way to Express Algorithms

- Two Main Classes of Programming Languages:

Prescriptive = How Computation must behave

- Imperative Languages: State + Mutable Value + Assignment + Sequence Control

Descriptive = What Computation must produce

- Declarative Languages: Immutable Value + Application + Composition

Programming Languages Paradigms

The main ones:

- **Procedural**: Fortran, Cobol, Algol, Pascal, C, ADA, ...
- **Functional**: Lisp, Scheme, ML, Haskell, OCAML, ...
- **Algebraic**: Lucid, OBJ, OPAL, ActOne
- **Logic, Constraint-based**: Prolog, LogLisp, Datalog, Parlog (SQL, spreadsheets languages,)...
- **Object Oriented**: Simula67, SmallTalk, C++, OCAML, Java, C#, F#, ...
- **Scripting**: Perl, Python, PHP, JavaScript...
- **Concurrency**: Lucid, OCCAM, C-Linda, PrologLinda, SPARK, Parlog, Java, C#, ...
- **Dataflow**: Lucid, C-Linda, PrologLinda,...
- **Multi-paradigms**: most of the most recent ones

50 years of Programming Languages

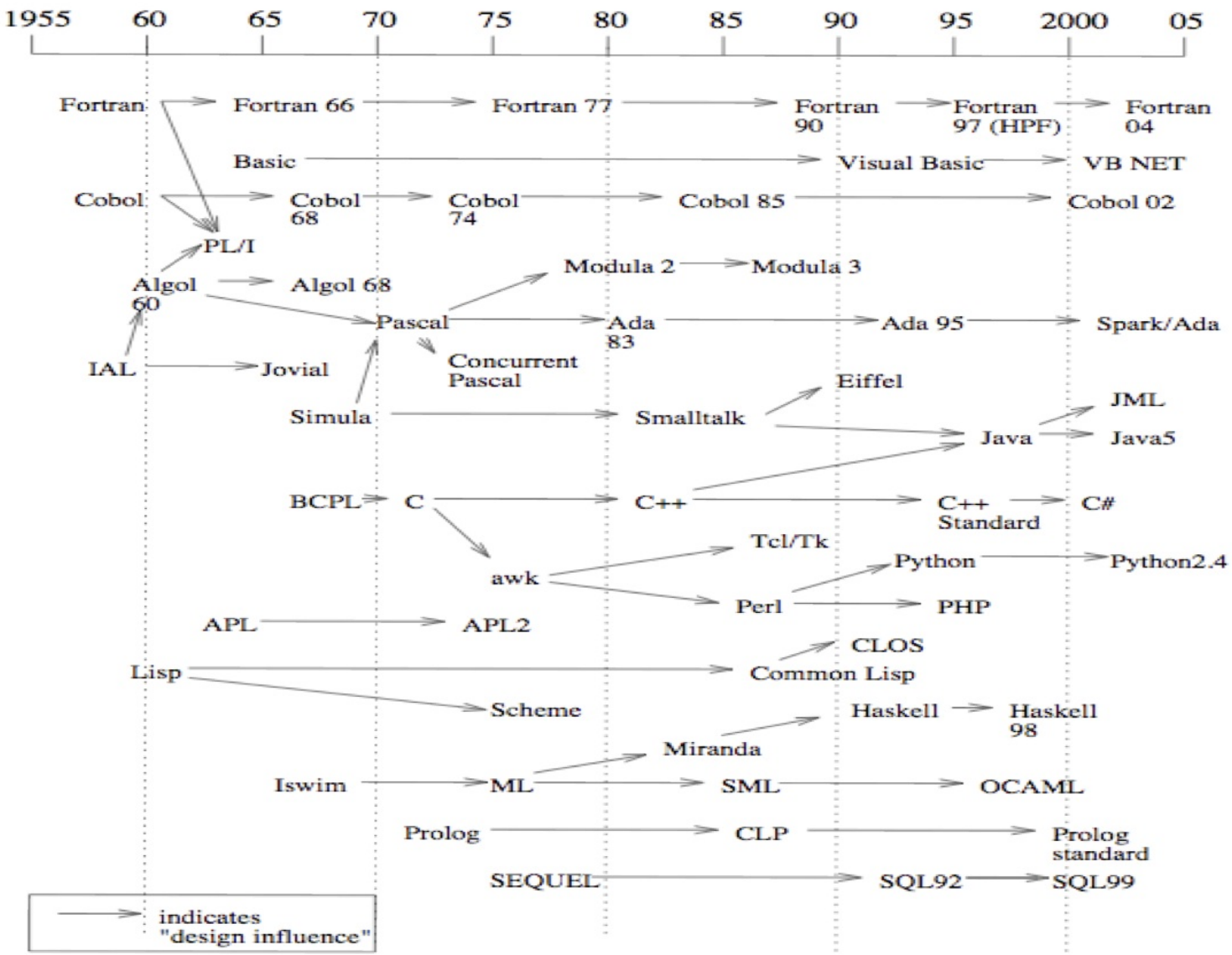

Figure 1.2: A Snapshot of Programming Language History

Overview of the Module

- **Specific Aim.** The knowledge of the main Programming Paradigms (including, Procedural, Functional, Logic, and Object, for sure) from the following viewpoints:
  - **Syntax and Semantics**: **F**ormalization of the costructs
  - **Programming Methodologies**: **U**se of the constructs in the Program Development Approaches
  - **Supporting Structures**: **I**mplementation of constructs
  - ACRONYM: **FUI**

- **Material.**

`http://www.di.unipi.it/~bellia`
`http://didawiki.cli.di.unipi.it/doku.php/magistraleinformatica/plp/start`

Complementary Readings

Mitchell, J.C. Concepts in Programming Languages. Cambridge Univ. Press, 2007, 529p., ISBN 978-0-52178-098-8.
Gabrielli, M. and Martini, S. Programming Languages: Principles and Paradigms. Springer, 2010, 440p., ISBN 978-1-84882-913-8.
Scott, M.L., Programming Language Pragmatics, Elsevier-Morgan Kaufmann Pub., 2009, 944p., ISBN 978-0-12374-514-9