F# for (Computer) Science and Engineering

Prof. Antonio Cisternino
University of Pisa

17/7/2012
Introduction

Increasing complexity

Machine programming → Algorithms → Software Coordination

Traditional languages difficult to use for non-computer scientists
BASIC was advertised as the language of the future
Matlab and Excel...

- ...have become the tools many scientists use every day
- Dynamic languages (i.e. Python) are used too
- Some scientists still use C/C++/Fortran code BUT they are decreasing in numbers
- Why?
- Software and computers are not anymore made only by/for science and engineering
F#

- .NET language
- Interoperates with C#/C++
- Type inference
- Has an interactive evaluation
- Efficient
- Feels like Python and, with VS, like Matlab
VSLab

- Add-in for Visual Studio
- Allows to interactively define and create VS toolwindows
- Enables Visual Studio to be extended interactively
- Matlab-like interaction (using Math.NET or other math library)
- Available on Visual Studio Shell
Teaching F# to scientists and engineers

- Started in 2008 with VSLab
- Collaboration with «Centro E. Piaggio» at University of Pisa
- Graduate and master students from Biomedical engineering course
- Goal: give to engineers a tool like Matlab (with less math libraries) capable of:
  - Interactive evaluation
  - Easy interaction with devices
  - Shipping standalone program eventually
Another useful lesson

- 2011: course on scientific programming at Scuola Normale Superiore (class of Science)
- Topics discussed interactively in Visual Studio
Just arrived...
And for Computer Science?

- ML has been used as first programming language by many CS courses in the past
- At UniPisa we *come back* from Java to ML 3yrs ago
  - F# was a driver (industrial support)
  - Good for start programming (top level)
  - Rich of core programming concepts
- Use of F# in UI programming course
  - Interactive interface building
  - Access to real world UI toolkits (WinForms, WPF, GTK#, ...)

Conclusions

- Functional programming (and F# syntax) is accepted by scientists and engineers quickly
- The OO programming constructs are easily accepted in the use BUT writing classes is a barrier
- F# has proven to be a good common language between computer scientists and scientists/engineering