© 2007 xkcd

COS 326 Functional Programming:
An elegant weapon for the modern age

David Walker

Princeton University

In 1936, Alonzo Church invented the lambda calculus. He called it a logic, but it was a language of pure functions -- the world's first programming language.

He said:

"*There may, indeed, be other applications of the system than its use as a logic.*"

Alonzo Church, 1903-1995
Princeton Professor, 1929-1967

Greatest technological understatement of the 20th century?

He said:

"*There may, indeed, be other applications of the system than its use as a logic.*"

Alonzo Church, 1903-1995
Princeton Professor, 1929-1967

Alonzo Church
1934 -- developed lambda calculus

Alan Turing (PhD Princeton 1938)
1936 -- developed Turing machines

Programming Languages

Computers

Optional reading: ***The Birth of Computer Science at Princeton in the 1930s***
by Andrew W. Appel, 2012. http://press.princeton.edu/chapters/s9780.pdf

A few designers of functional programming languages

Alonzo Church:
λ-calculus, 1934

John McCarthy
(PhD Princeton 1951)
LISP, 1958

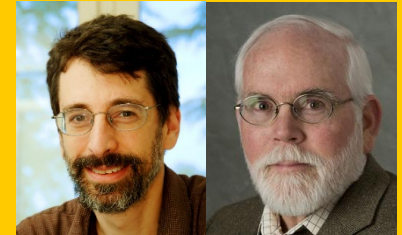Guy Steele & Gerry Sussman:
Scheme, 1975

Robin Milner
ML, 1978

Luca Cardelli
Edinburgh ML, 1981

Standard ML
1986

CaML
1980s

Implementations:

Appel & MacQueen: SML/NJ, 1988

Xavier Leroy:
Ocaml, 1990's

They were younger than they appear…

Alonzo Church:
λ-calculus, 1934
Photo ~1960

John McCarthy
(PhD Princeton 1951)
LISP, 1958
Photo ~1975

Guy Steele & Gerry Sussman:
Scheme, 1975

Photo ~1995   Photo ~1995

Robin Milner
ML, 1978
Photo ~2005

Luca Cardelli
Edinburgh ML, 1981
Photo ~1995

Implementations:
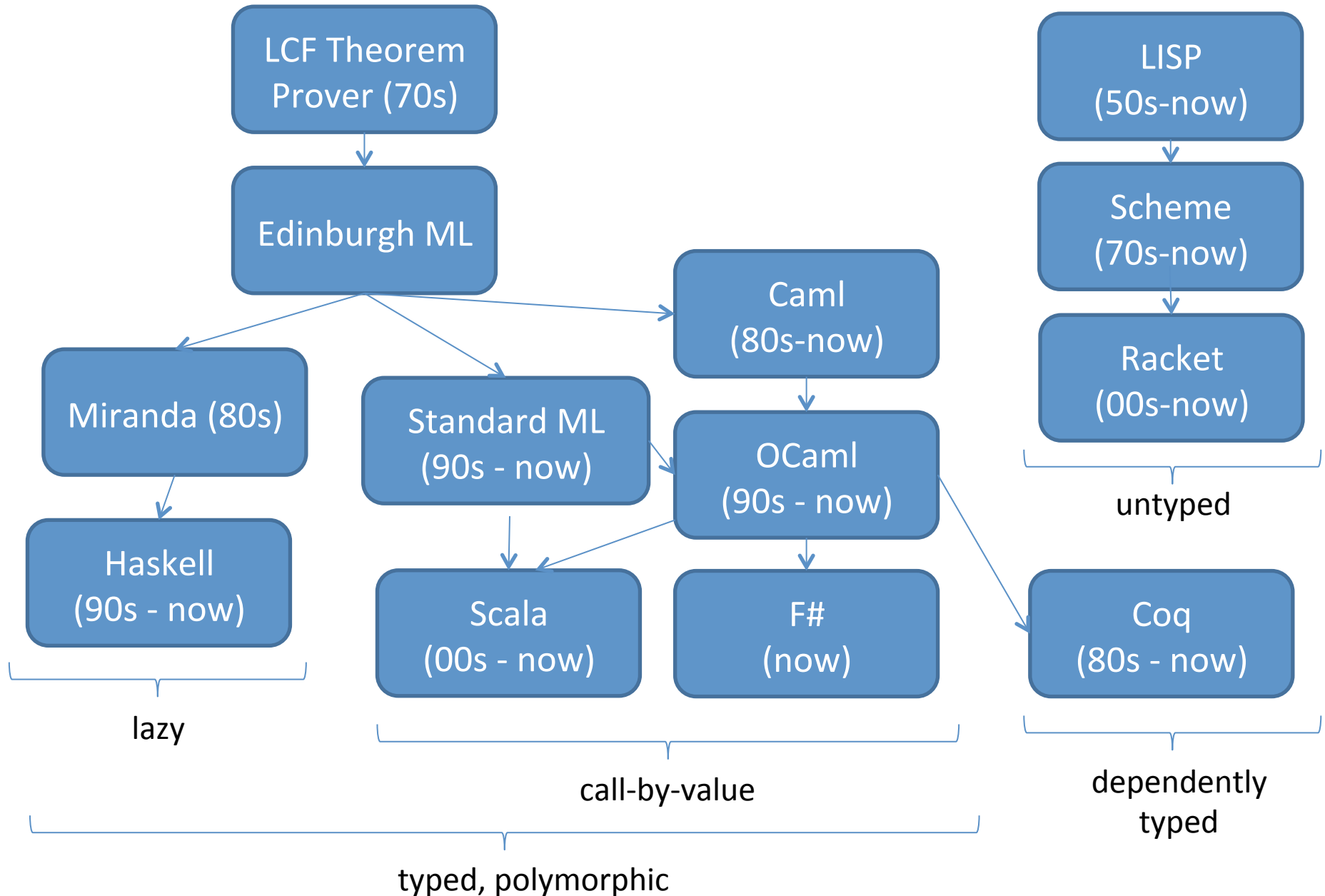

Appel & MacQueen: SML/NJ, 1988
Photo 2005   Photo ~2000

Xavier Leroy:
Ocaml, 1990's

Photo ~2005

Vastly Abbreviated FP Geneology

LCF Theorem Prover (70s)

Edinburgh ML

Caml (80s-now)

Miranda (80s)

Standard ML (90s - now)

OCaml (90s - now)

Haskell (90s - now)

Scala (00s - now)

F# (now)

Coq (80s - now)

LISP (50s-now)

Scheme (70s-now)

Racket (00s-now)

lazy

call-by-value
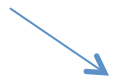
typed, polymorphic

untyped

dependently typed

But Why Functional Programming *Now*?

- Functional programming will introduce you to new ways to *think about* and *structure* your programs:
  - new reasoning principles
  - new abstractions
  - new design patterns
  - new algorithms
  - elegant code

- Technology trends point to increasing parallelism:
  - multicore, gpu, data center
  - functional programming techniques such as map-reduce provide a plausible way forward for many applications

Functional Languages: Who's using them?

map-reduce in their data centers

Scala for
correctness, maintainability, flexibility

F# in Visual Studio

Erlang for
concurrency,
Haskell for
managing PHP

mathematicians

Haskell to
synthesize hardware

Coq (re)proof of
4-color theorem

www.artima.com/scalazine/articles/twitter_on_scala.html
gregosuri.com/how-facebook-uses-erlang-for-real-time-chat
www.janestcapital.com/technology/ocaml.php
msdn.microsoft.com/en-us/fsharp/cc742182
labs.google.com/papers/mapreduce.html
www.haskell.org/haskellwiki/Haskell_in_industry

Haskell
for specifying
equity derivatives

Functional Languages: Join the crowd

- Elements of functional programming are showing up all over
  - F# in Microsoft Visual Studio
  - Scala combines ML (a functional language) with Objects
    - runs on the JVM
  - C# includes "delegates"
    - delegates == functions
  - Python includes "lambdas"
    - lambdas == more functions
  - Javascript
    - find tutorials online about using functional programming techniques to write more elegant code
  - C++ libraries for map-reduce
    - enabled functional parallelism at Google
  - Java has generics and GC
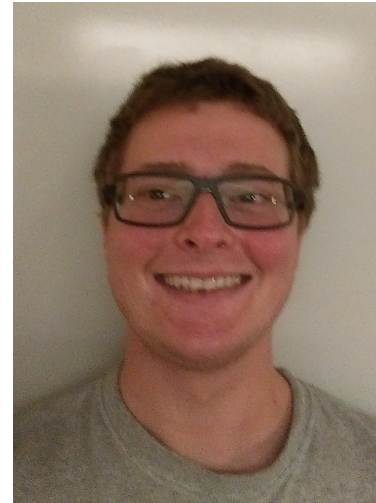 - …

COURSE LOGISTICS

Course Staff

David Walker
Professor
office: COS 211
email: dpw@

Christopher Moretti
Teaching Faculty
Head Preceptor
office: COS 208
email: cmoretti@

Anders Miltner
Grad Student
office: Fishbowl
email: amiltner@cs

Robin Qiu
Grad Student
office: Fishbowl
email: yqiu@cs

Resources

- Web:
  - http://www.cs.princeton.edu/~cos326

- Lecture schedule and readings:
  - $(coursehome)/lectures.php

- Assignments:
  - $(coursehome)/assignments.php

- Precepts
  - useful if you want to do well on exams and homeworks

- Install OCaml: $(coursehome)/resources.php

Collaboration Policy

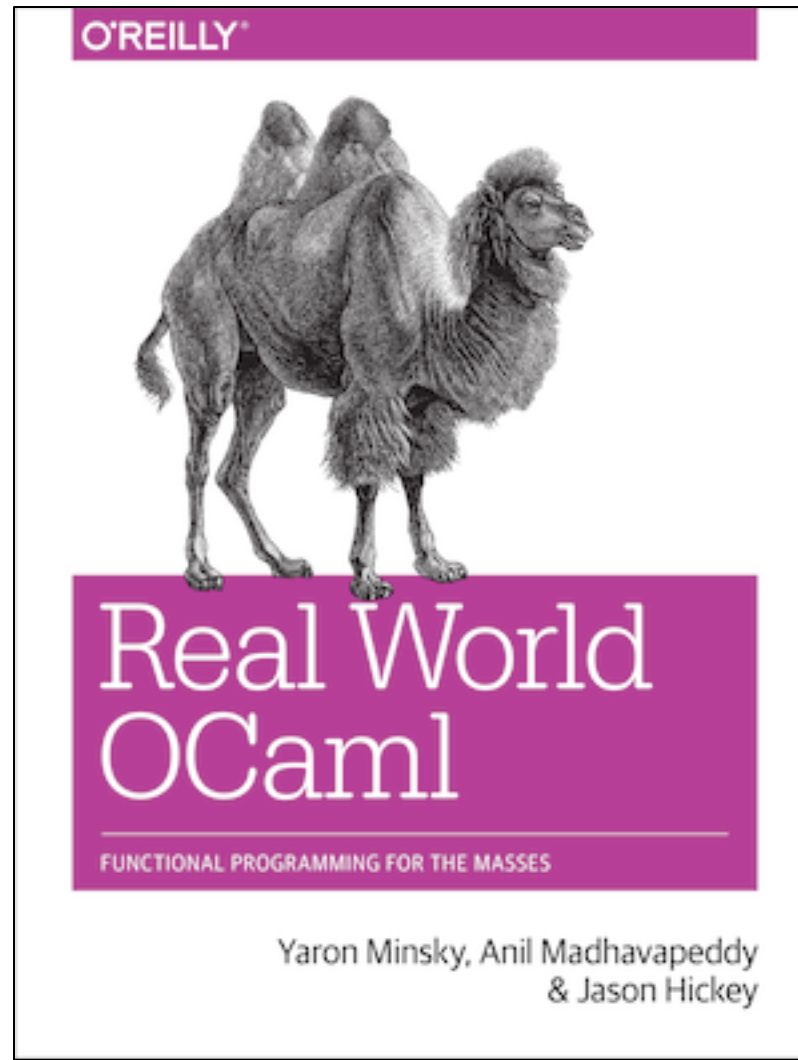The COS 326 collaboration policy can be found here:

http://www.cs.princeton.edu/courses/archive/fall13/cos326/info.php#collab

Read it in full prior to beginning the first assignment.

Please ask questions whenever anything is unclear, at any time during the course.

Course Textbook

http://realworldocaml.org/

Final Exam

There will be a midterm exam, in midterm week

Wednesday, October 26

There will be a final exam, in exam period

(January — Make your travel plans accordingly)

Assignment 0

Figure out how to download and install OCaml 4.03
on your machine by the time precept begins tomorrow.
(or, how to use OCaml by ssh to Princeton University servers)

Resources Page:

http://www.cs.princeton.edu/courses/archive/fall15/cos326/resources.php

Hint:

ocaml.org

Public Service Announcement

The Pen is Mighter than the Keyboard: Advantages of Longhand Over Laptop Note Taking

Pam Mueller (Princeton University)
Daniel Oppenheimer (UCLA)
Journal of Psychological Science, June 2014, vol 25, no 6

http://pss.sagepub.com/content/25/6/1159.fullkeytype=ref&siteid=sppss&ijkey=CjRAwmrlURGNw

- You learn conceptual topics better by taking notes by hand.
- Facebook and World of Warcraft distract your classmates.