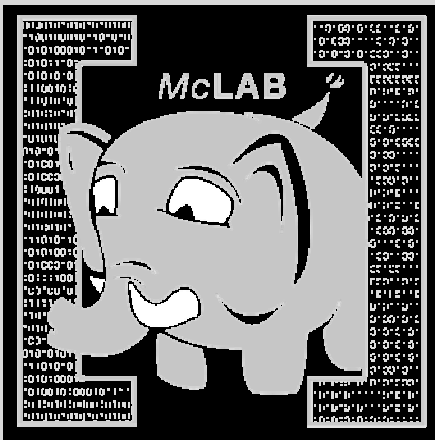


McLab Tutorial

www.sable.mcgill.ca/mclab

Laurie Hendren, Rahul Garg and
Nurudeen Lameed



Other McLab team members:

Andrew Casey, Jesse Doherty, Anton Dubrau, Jun Li,
Amina Aslam, Toheed Aslam, Maxime Chevalier-
Boisvert, Soroush Radpour, Oliver Savary, Maja
Frydrychowicz, Clark Verbrugge

Sable Research Group
School of Computer Science
McGill University, Montreal, Canada

Tutorial Overview

- Why MATLAB?
- Introduction to MATLAB – challenges
- Overview of the McLab tools
 - Introduction to the front-end and extensions
 - IRs, Flow analysis framework and examples
 - Back-ends including the McVM virtual machine
- Wrap-up

Nature Article: “Why Scientific Computing does not compute

- 38% of scientists spend at least 1/5th of their time programming.
- Codes often buggy, sometimes leading to papers being retracted. Self-taught programmers.
- Monster codes, poorly documented, poorly tested, and often used inappropriately.
- 45% say scientists spend more time programming than 5 years ago.



FORTTRAN
C/C++

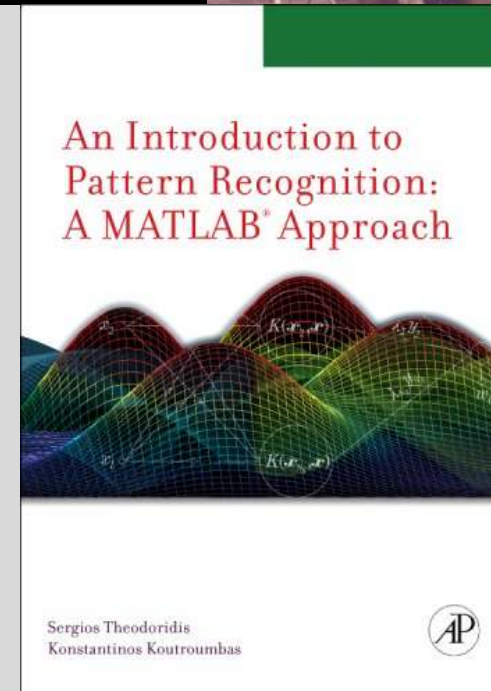
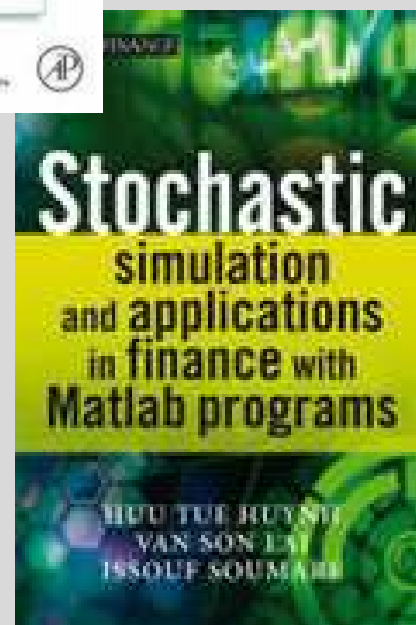
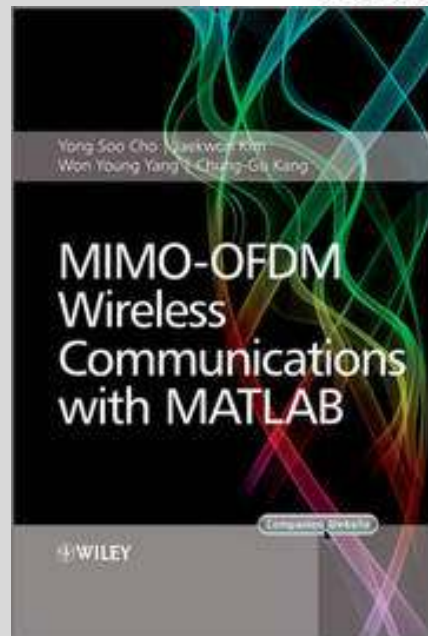
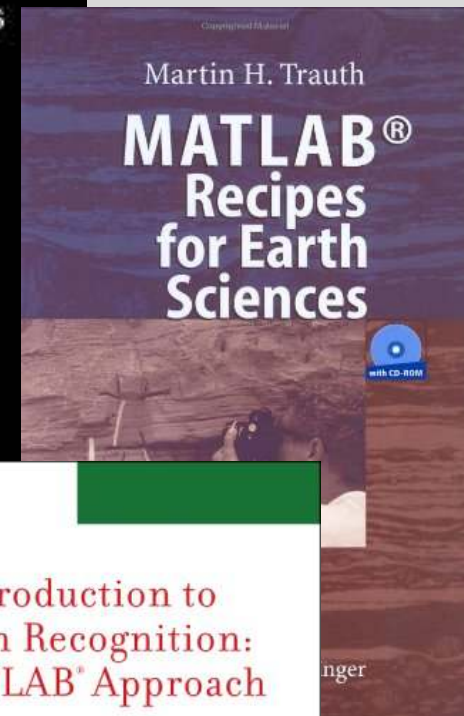
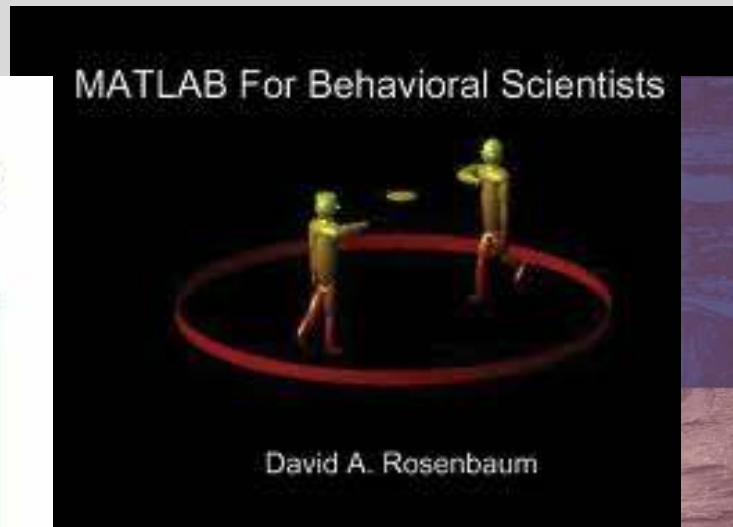
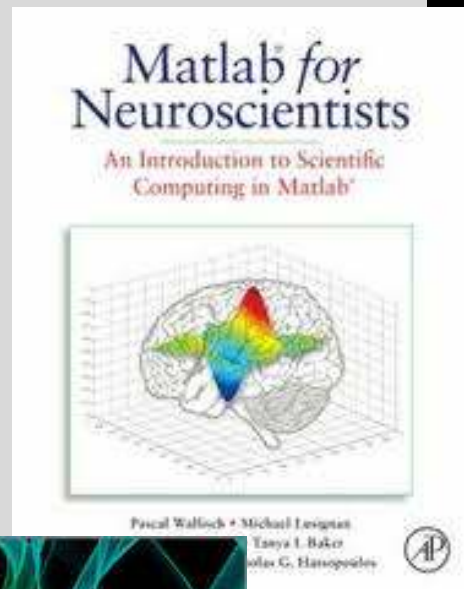
Java
AspectJ



MATLAB
PERL
Python
Domain-specific

A lot of MATLAB programmers!

- Started as an interface to standard FORTRAN libraries for use by students.... but now
 - 1 million MATLAB programmers in 2004, number doubling every 1.5 to 2 years.
 - over 1200 MATLAB/Simulink books
 - used in many sciences and engineering disciplines
- Even more “unofficial” MATLAB programmers including those using free systems such as Octave or SciLab.



Why do Scientists choose MATLAB?

REASONS WHY PEOPLE WHO WORK WITH COMPUTERS SEEM TO HAVE A LOT OF SPARE TIME... eviljays.com

Web Developer



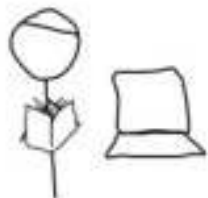
'Its uploading'

Sysadmin



'Its rebooting'

3D Artist



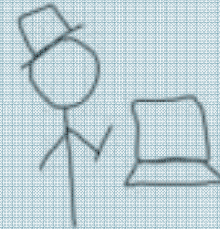
'Its rendering'

IT Consultant



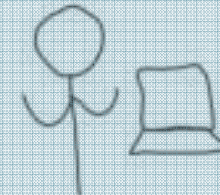
'Its your problem now'

Hacker



'Its scripted'

Programmer



'Its compiling'

MATLAB



FORTTRAN



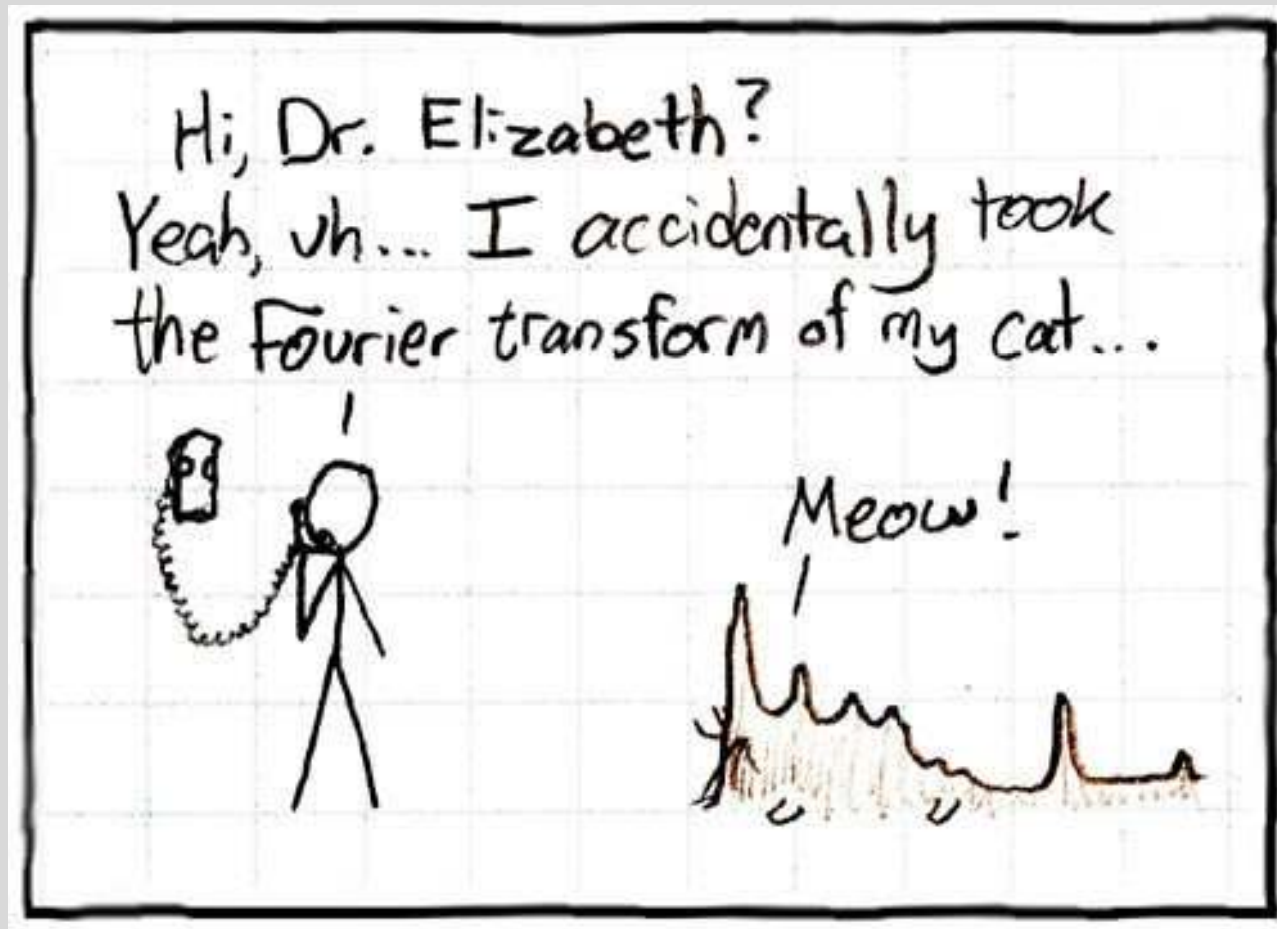
Implications of choosing a dynamic, “scripting” language like MATLAB....



Interpreted ...

**Potentially large
runtime
overhead in
both time and
space**

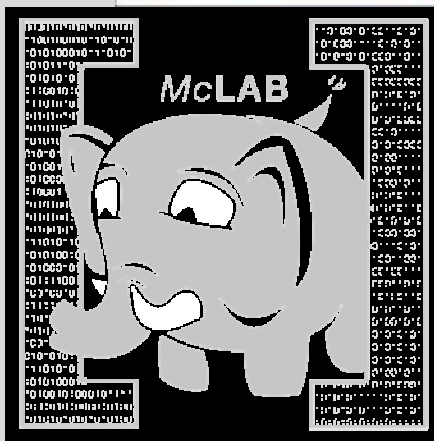
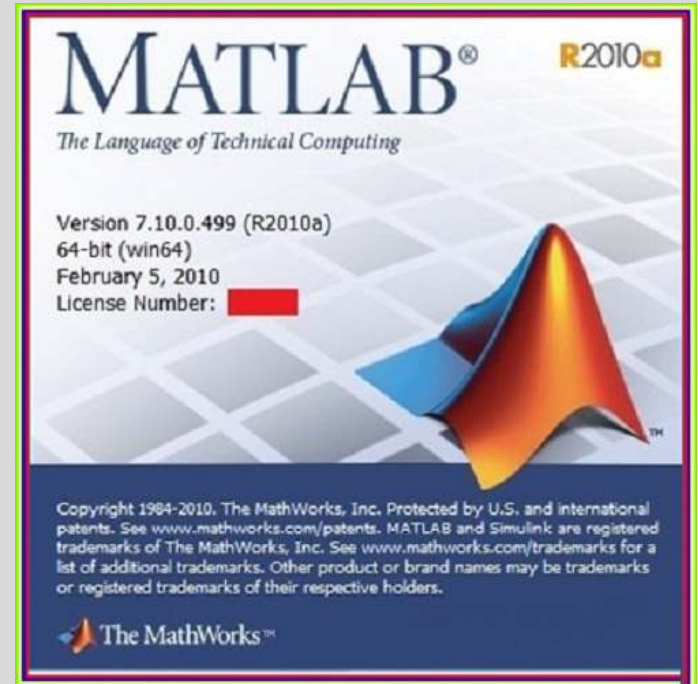
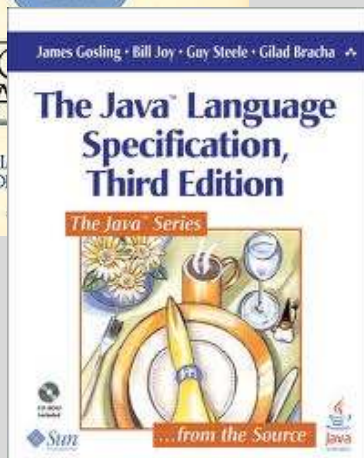
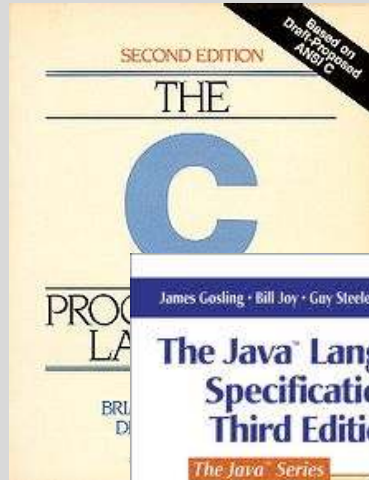
No types and “flexible” syntax



Most semantic (syntactic) checks made at runtime ... No static guarantees



No formal standards for MATLAB



Culture Clash

Scientists / Engineers

- Comfortable with informal descriptions and “how to” documentation.
- Don’t really care about types and scoping mechanisms, at least when developing small prototypes.
- Appreciate libraries, simple tool support, and interactive development tools.

Programming Language / Compiler Researchers

- Prefer more formal language specifications.
- Prefer well-defined types (even if dynamic) and well-defined scoping and modularization mechanisms.
- Appreciate “harder/deeper/more beautiful” research problems.

Goals of the McLab Project

- Improve the understanding and documentation of the semantics of MATLAB.
- Provide front-end compiler tools suitable for MATLAB and language extensions of MATLAB.
- Provide a flow-analysis framework and a suite of analyses suitable for a wide range of compiler/soft. eng. applications.
- Provide back-ends that enable experimentation with JIT and ahead-of-time compilation.

Overview of McLab/Tutorial

