

# Programming Languages

Members:

Bruno C. d. S. Oliveira ([bruno@cs.hku.hk](mailto:bruno@cs.hku.hk)) (web: <http://i.cs.hku.hk/~bruno/>)

Jeremy Bi

Tomas Tauber

Yanlin Wang

Linus Yang

Haoyuan Zhang

Weixin Zhang

Jimmy Zhao

Ningning Yi

# Aim of Programming Languages

The key aim of Programming Languages is to increase **Programmer's Productivity**

Programming Language Research aims at:

- Allowing faster development cycles
- Supporting Large-Scale Programming
- Preventing bugs

By creating new Programming Languages & Abstractions



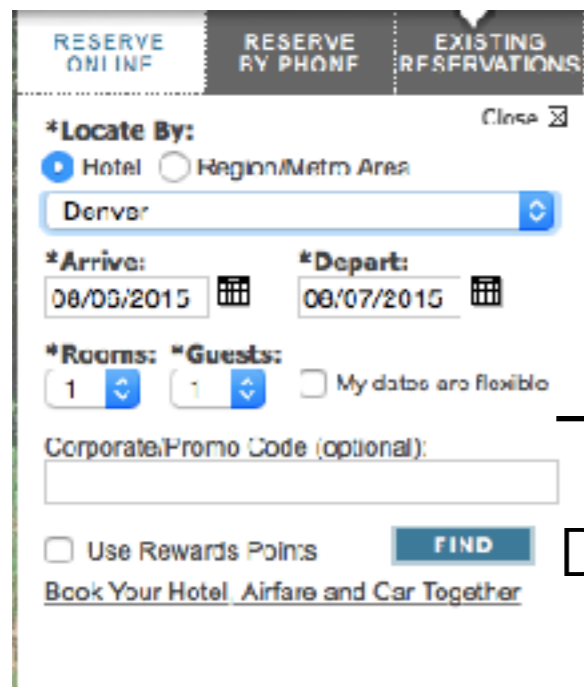
Why do we need so many  
languages?

# Software Engineering

Web

Server

DB



A screenshot of a hotel reservation web form. The form has a dark header with three tabs: 'RESERVE ONLINE', 'RESERVE BY PHONE', and 'EXISTING RESERVATIONS'. Below the header, there are several sections: '\*Locate By:' with radio buttons for 'Hotel' (selected) and 'Region/Metro Area', and a dropdown menu showing 'Denver'; '\*Arrive:' and '\*Depart:' fields with dates '08/03/2015' and '08/07/2015' respectively; '\*Rooms:' and '\*Guests:' dropdowns both set to '1', with a checkbox for 'My dates are flexible'; a text input field for 'Corporate/Promo Code (optional)'; a checkbox for 'Use Rewards Points'; a blue 'FIND' button; and a link 'Book Your Hotel, Airfare and Car Together'.



Data Format (XML/JSON/...)

PhP/Html/...

Java/C/...

SQL

# Why do we need so many languages?

Possible Answers:

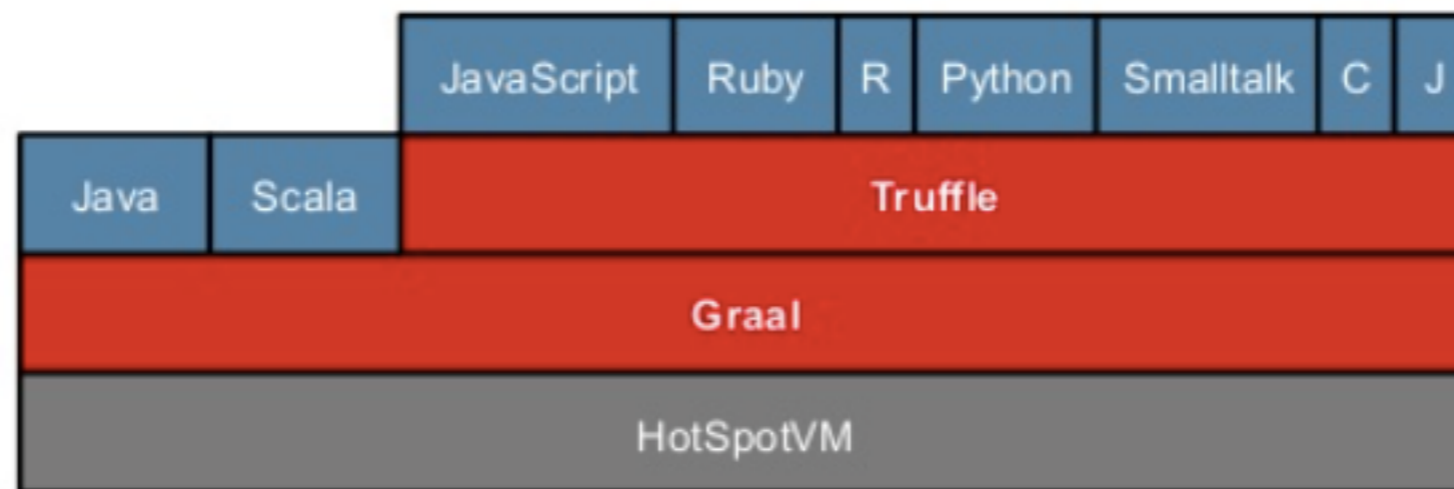
- 1) Existing languages have limitations;
- 2) We need specialised languages for a certain domain
  - Queries;
  - GPU Programming;
  - Parsing;
  - Build Systems;
  - Domain-specific languages
  - ...

# Research Topic 1: Rapid Language Development

- Clearly we often need to develop new languages
- However, building new languages that are **correct**, **robust**, and **efficient** is very hard and time consuming!
- How can we solve this problem? **We should have ways to do rapid and easy language development!**
- **Goal:** Allow programmers which are not experts in compilers to build their own languages.

# Research Topic 1: Rapid Language Development

Meet Truffle and Graal from **Oracle**



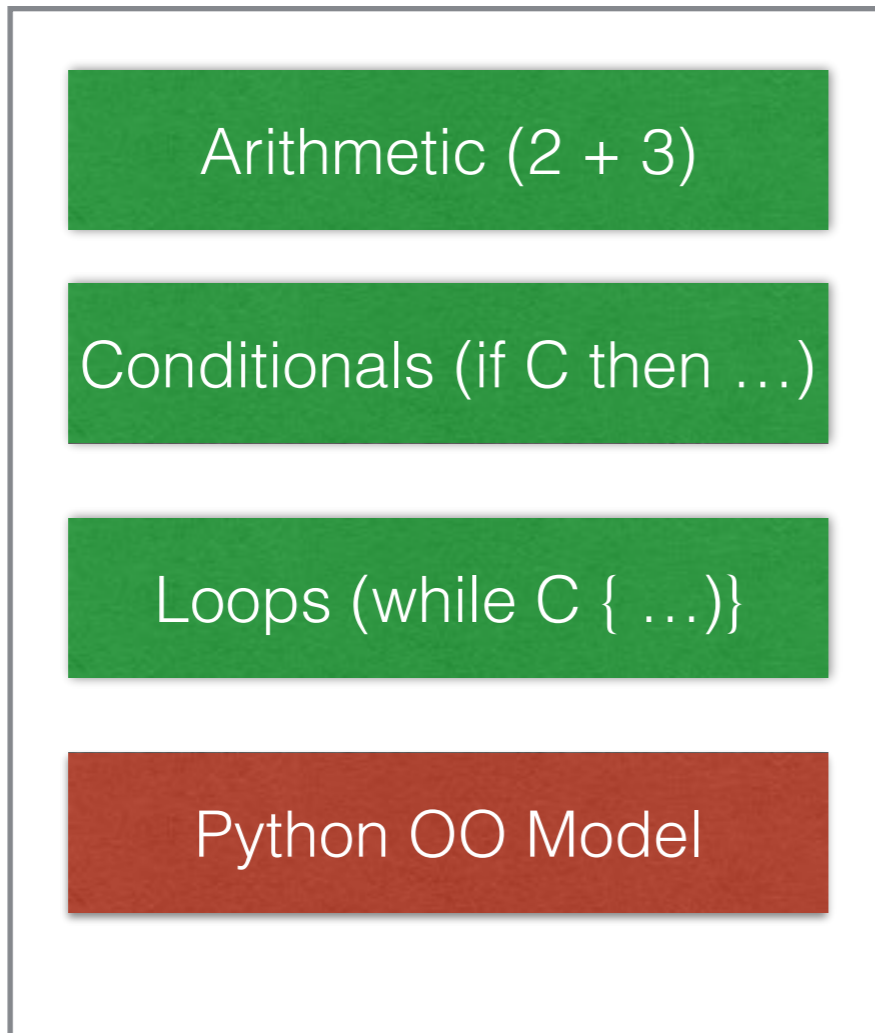
They allow you to develop **efficient** language implementations by creating **interpreters**.

But we still need to build languages from scratch...

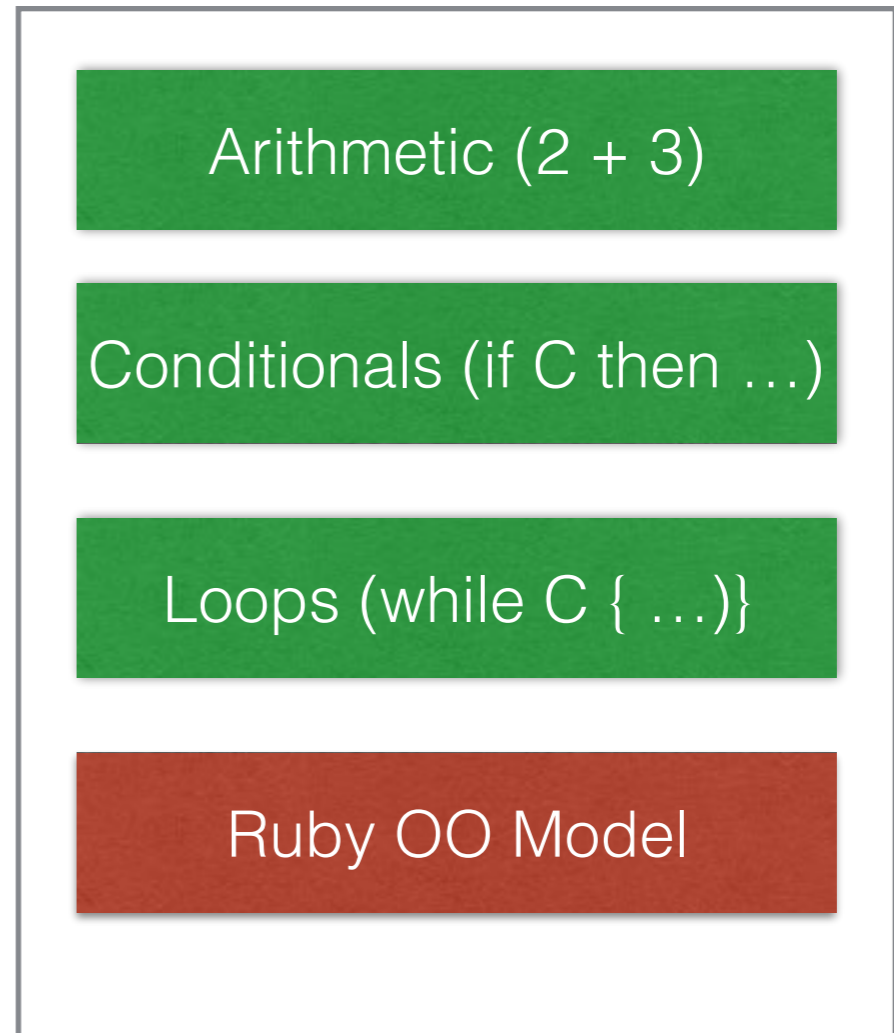


# Research Topic 1: Rapid Language Development

Python



Ruby



Observation: Languages have a lot in common!

# Research Topic 1: Rapid Language Development

- We are developing a new Java-based Framework for Language Components on top of Graal and Truffle
- This framework uses some of the latest results in software modularity and reuse, to modularize language components.

## References:

B. C. d. S. Oliveira and W. Cook. [Extensibility for the Masses](#), ECOOP 2012

B. C. d. S. Oliveira et al. [Feature-Oriented Programming with Object Algebras](#), ECOOP 2013

Haoyuan Zhang et al. [Scrap your Boilerplate with Object Algebras](#), OOPSLA 2015

# Research Topic 2: Functional Programming

- The world is getting functional!
- Lambdas (first-class functions) in Mainstream Languages:
  - Java 8, C++11, C#
- Swift, Apple's new language is designed as a functional language



# Research Topic 2: Functional Programming

- Current mainstream programming languages are typically Imperative and Object-Oriented.
- However, sometimes traditional imperative programming leads to complex solutions, abuse of mutable state and bugs!

# Research Topic 2: Functional Programming

- Functional Programming is a different programming style that avoids the use of mutable state and encourages reuse through first class functions
- Quicksort in a Functional Style:

```
quicksort [] = []
quicksort (x:xs) =
    let smallerSorted = quicksort [a | a <- xs, a <= x]
        biggerSorted = quicksort [a | a <- xs, a > x]
    in smallerSorted ++ [x] ++ biggerSorted
```