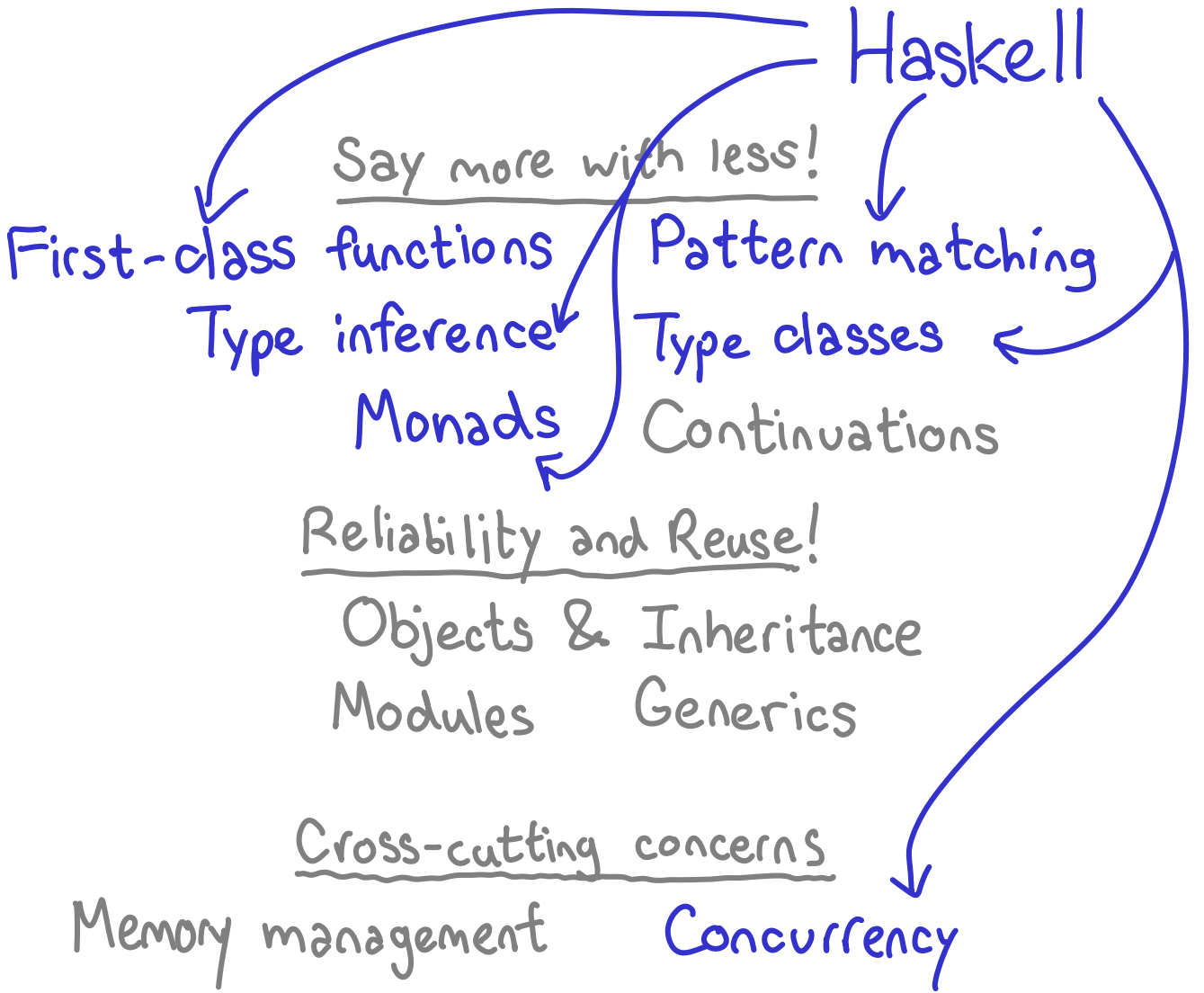
CSCI-UA 490

Haskell

Edward Z. Yang


Haskell
Say more with less!
First-class functions
Pattern matching
Type inference
Type classes
Monads
Continuations
Reliability and Reuse!
Objects & Inheritance
Modules
Generics
Cross-cutting concerns
Memory management
Concurrency

What is Haskell?

a typed, lazy, pure, functional programming language

What is Haskell?

functions first class

expressions over instructions

a typed, lazy, pure, functional programming language

What is Haskell?

No mutation/side effects
Functions do the same thing

a typed, lazy, pure, functional programming language

equational reasoning

parallelism

maintainability

What is Haskell?

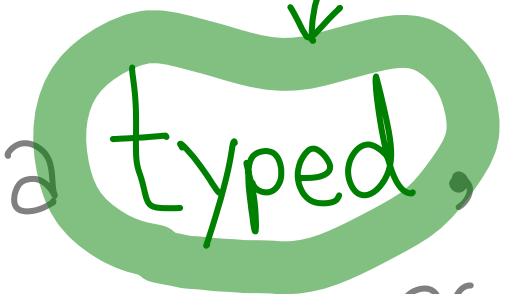
Expressions not evaluated until needed

Custom control structures

Infinite data structures

Compositionality

What is Haskell?



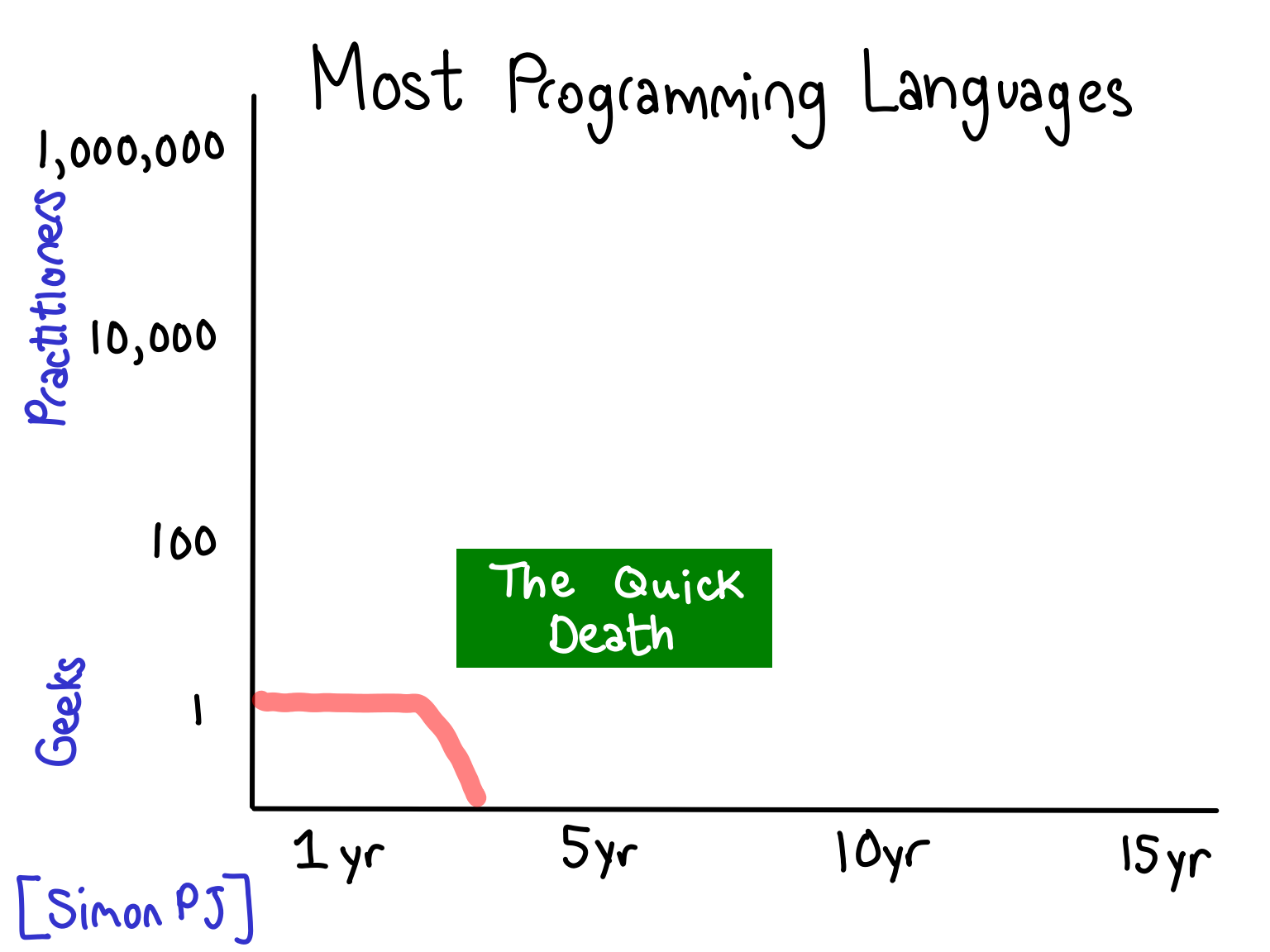
more expressive than what you might be used to!

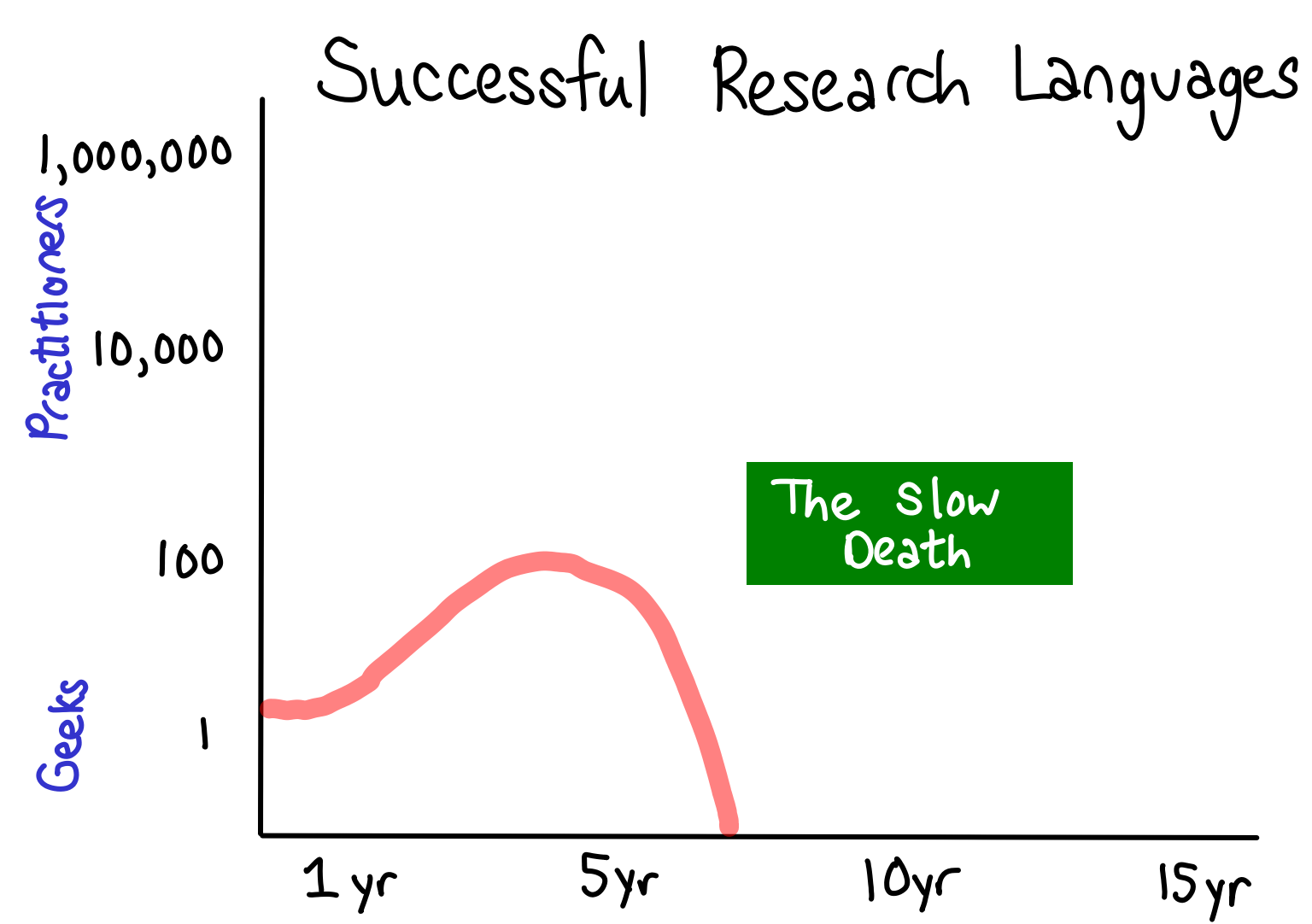
Why Haskell?

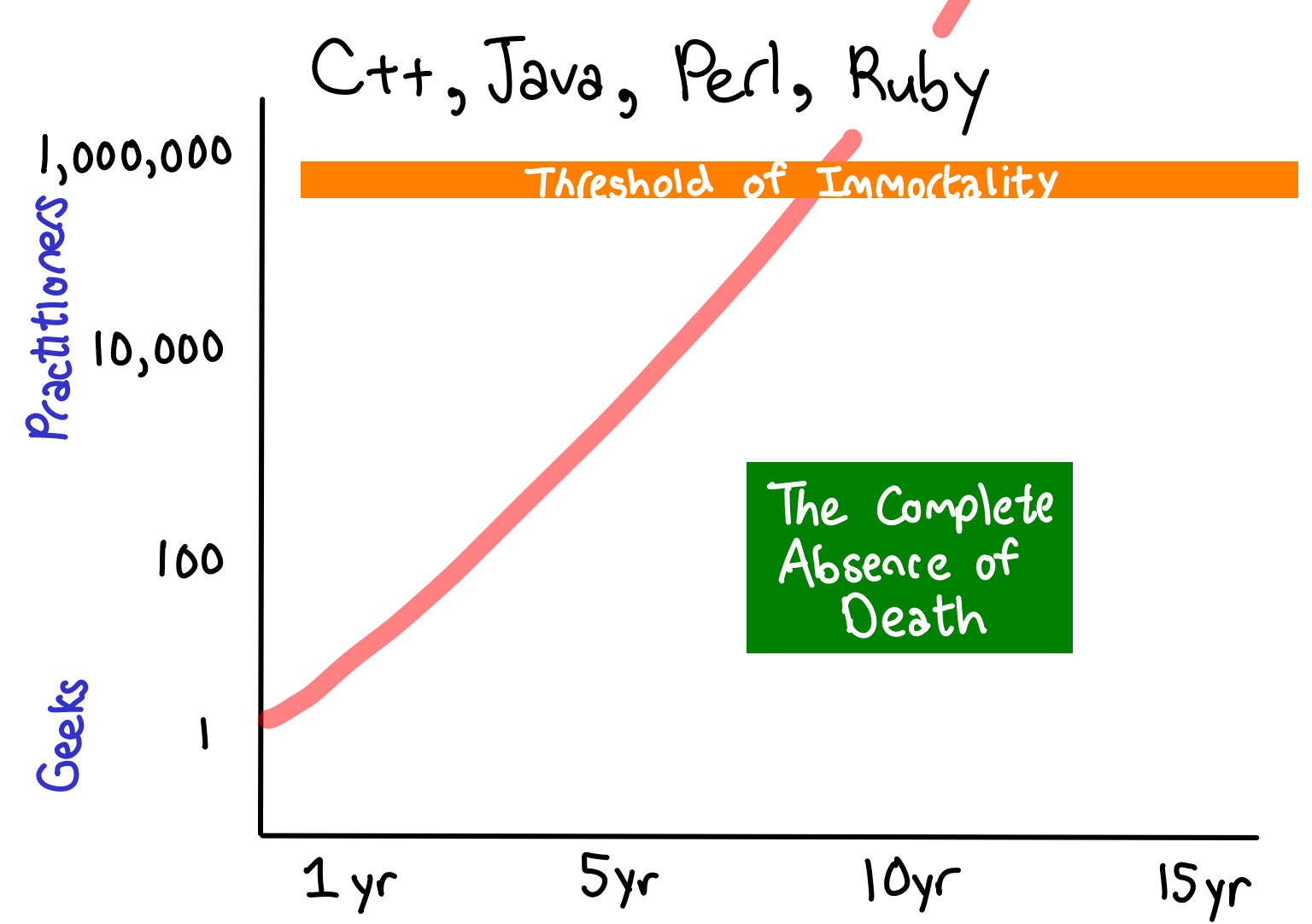
Types will change the way you think

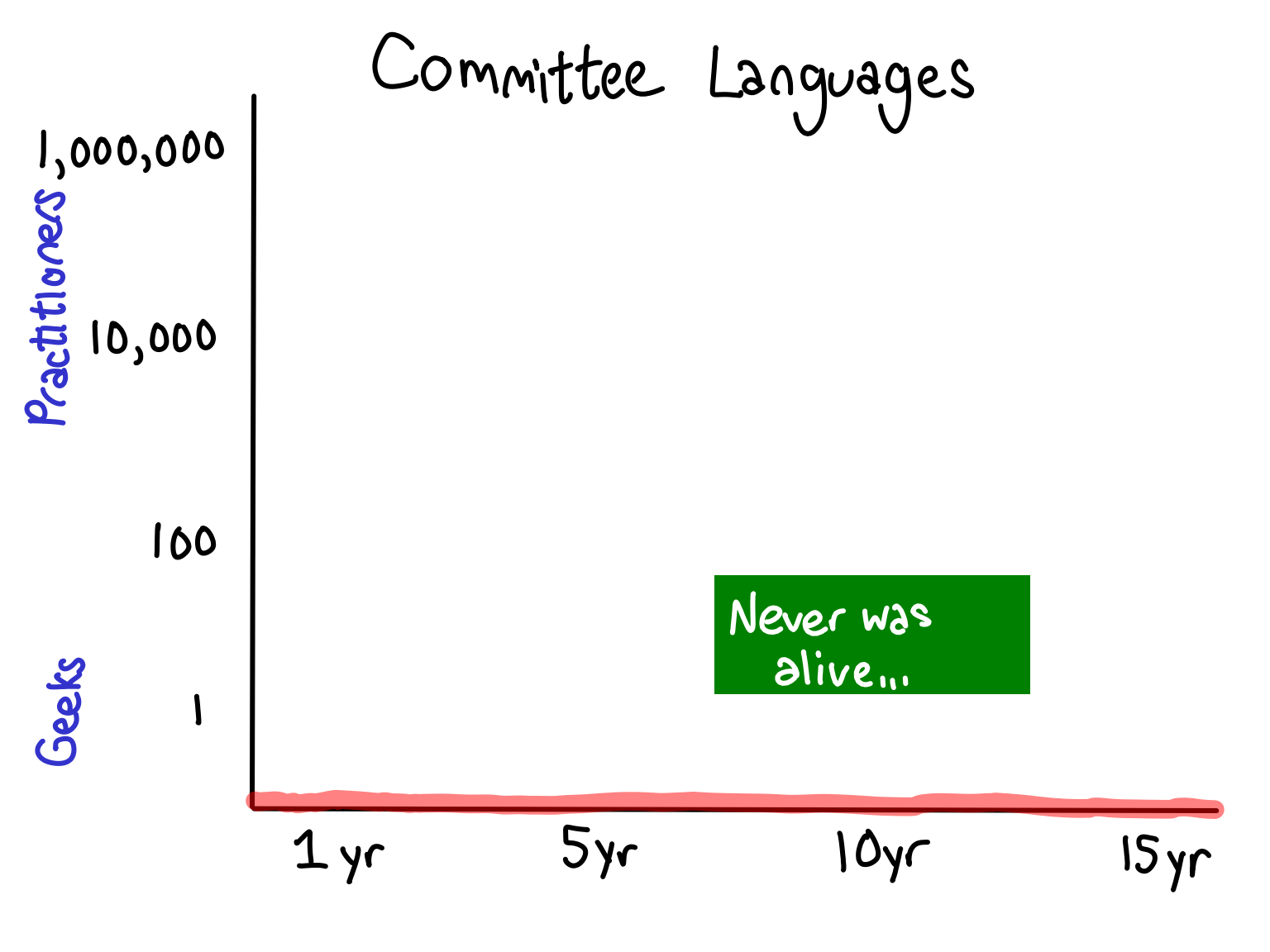
Raise the level of abstraction

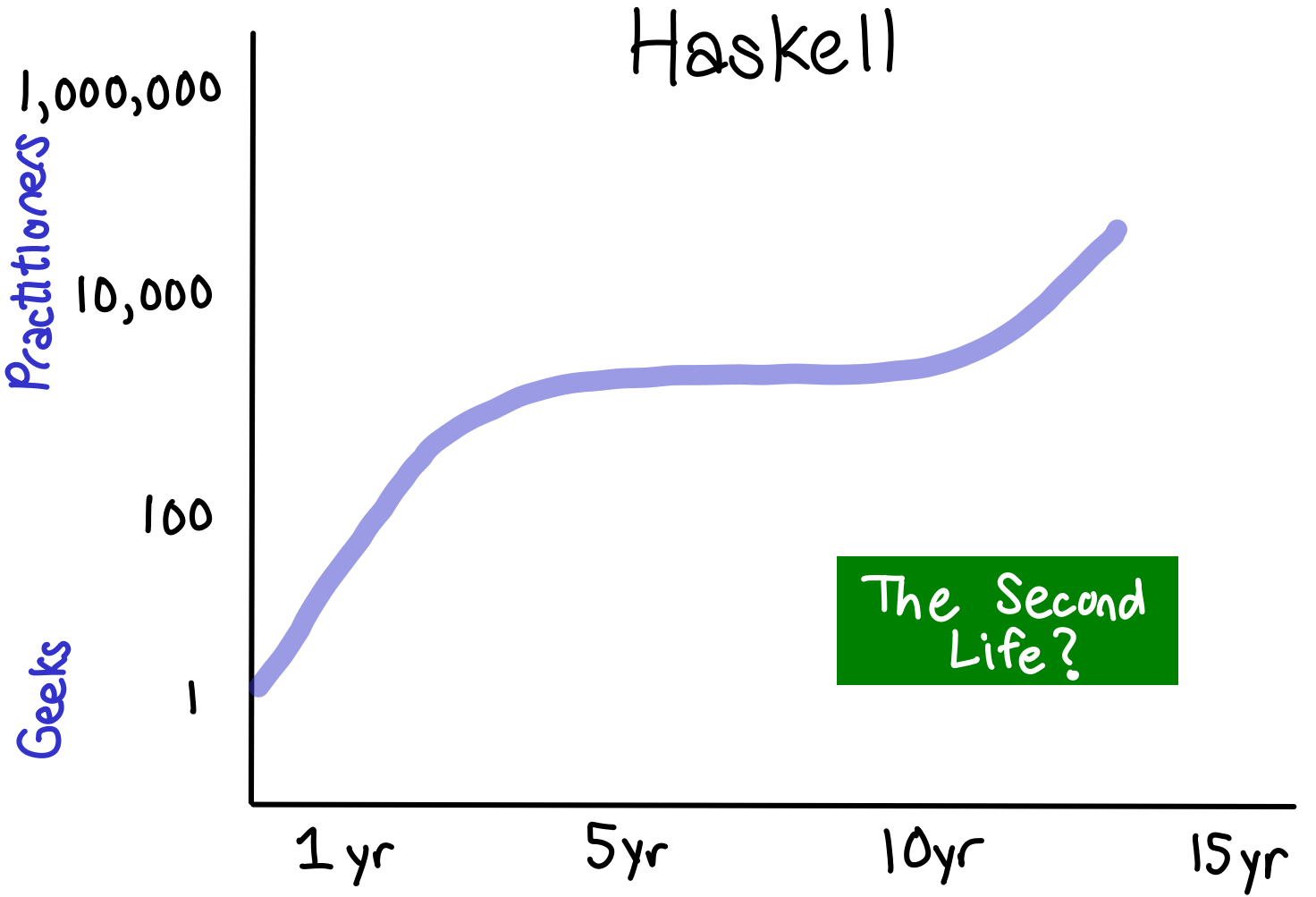
Purity to be essential for multicore


Most Programming Languages
1,000,000
10,000
100
1
Practitioners
Geeks
The Quick Death
1 yr
5yr
10yr
15yr
[Simon PJ]


Successful Research Languages
1,000,000
10,000
100
1
Practitioners
Geeks
The Slow Death
1 yr
5yr
10yr
15yr


C++, Java, Perl, Ruby
1,000,000
Threshold of Immortality
Practitioners
10,000
100
The Complete Absence of Death
Geeks
1
1 yr
5yr
10yr
15yr


Committee Languages
1,000,000
10,000
100
1
Practitioners
Geeks
Never was alive...
1 yr
5yr
10yr
15yr


Haskell
1,000,000
10,000
100
1
Practitioners
Geeks
1 yr
5yr
10yr
15yr
The Second Life?

Announcements

Homework 1 is out (see Course Website)

due Monday! (get started!)